



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
Área Departamental de Engenharia de Sistemas de Potência e
Automação

Ramo Energia

**Desenvolvimento de algoritmos para a
determinação da máxima injeção nodal em redes
de energia elétrica**

João Gonçalo Matias Lopes Nunes

Dissertação para obtenção do Grau de Mestre em
Engenharia Electrotécnica – ramo de Energia

Orientador: Professor Doutor Francisco Alexandre Ganho da Silva Reis

Júri:

Presidente: Professor Jorge Alberto Mendes de Sousa

Vogais: Professor Francisco Alexandre Ganho da Silva Reis
Professor Mário Ventim Neves

Lisboa, Setembro 2012



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
Área Departamental de Engenharia de Sistemas de Potência e
Automação

Ramo Energia

**Desenvolvimento de algoritmos para a
determinação da máxima injeção nodal em redes
de energia elétrica**

João Gonçalo Matias Lopes Nunes

Dissertação para obtenção do Grau de Mestre em
Engenharia Electrotécnica – ramo de Energia

Orientador: Professor Doutor Francisco Alexandre Ganho da Silva Reis

Júri:

Presidente: Professor Jorge Alberto Mendes de Sousa

Vogais: Professor Francisco Alexandre Ganho da Silva Reis
Professor Mário Ventim Neves

Lisboa, Setembro 2012

AGRADECIMENTOS

Ao ISEL, docentes e colegas que possibilitaram a oportunidade de realização do mestrado, Engenharia Electrotécnica – Especialização em Energia, e desenvolvimento da presente tese.

Gostaria de expressar o meu profundo agradecimento ao meu professor e orientador da tese, Professor Doutor Eng.º Francisco Alexandre Ganho da Silva Reis, pela sua compreensão, encorajamento e disponibilidade ao longo deste projeto, por vezes a custo pessoal. O seu apoio e conhecimentos transmitidos fizeram com que fosse possível a sua realização.

Um especial agradecimento aos meus pais, à minha irmã e à Guida pela compreensão, paciência e apoio que me têm dado ao longo destes anos.

RESUMO

A presente dissertação apresenta um conjunto de algoritmos, cujo objetivo é a determinação da capacidade máxima de energia que é possível integrar numa rede de energia elétrica, seja num único nó ou em vários nós simultaneamente. Deste modo, obtém-se uma indicação dos locais mais adequados à nova instalação de geração e quais os reforços de rede necessários, de forma a permitirem a alocação da nova energia.

Foram estudados e identificados os fatores que influenciam o valor da capacidade máxima nodal, assim como as suas consequências no funcionamento da rede, em particular o carácter simultâneo associado às referidas injeções nodais. Nesse sentido, são apresentados e desenvolvidos algoritmos que têm em consideração as características técnicas da geração a ligar e as restrições físicas impostas pela rede elétrica existente. Os algoritmos desenvolvidos apresentados baseiam-se em busca gaussiana, tendo sido igualmente implementada uma heurística que tem em consideração a proximidade de outras injeções em nós adjacentes e finalmente, dada a natureza combinatória do problema, propõe-se a aplicação de algoritmos genéticos especificamente adaptados ao problema

Conclui-se que os algoritmos genéticos encerram características que lhes permitem ser aplicados em qualquer topologia com resultados superiores a todos os algoritmos desenvolvidos.

Os métodos apresentados foram desenvolvidos e implementados usando a linguagem de programação *Python*, tendo-se desenvolvido ainda um interface visual ao utilizador, baseado em *wxPython*, onde estão implementadas diversas ferramentas que possibilitam a execução dos algoritmos, a configuração dos seus parâmetros e ainda o acesso à informação resultante dos algoritmos em formato *Excel*.

Palavras-chave: máxima injeção nodal, algoritmo genético, geração distribuída

ABSTRACT

The present dissertation describes a set of algorithms which goal is to determine the maximum power that can be integrated into a power grid, either at a single node or at multiple nodes simultaneously. Thus obtaining an indication of the most suitable places to install new generation and which network reinforcements are necessary in order to permit the allocation of new energy.

Were studied and identified the factors that influence the value of the maximum nodal capacity, as well as their consequences on the electrical grid, in particular the simultaneously factor associated with the nodal injections. Following this analysis, are presented and developed algorithms that take into account the generation characteristics and the constraints imposed by the existing electrical grid. The algorithms developed are based on Gaussian search, heuristic search which takes into consideration the proximity to other adjacent nodes injections and finally, given the combinatorial nature of the problem, it is proposed the application of genetic algorithms specifically adapted to problem.

It is concluded, that the genetic algorithms have characteristics that allow them to be applied to any topology with better results than the other algorithms developed here.

These methods were developed and implemented using the *Python* programming language, having also developed a Graphic User Interface (GUI), based on *wxPython*. This GUI has implemented several tools that enable the execution of the algorithms, the setting of its parameters and the access to the information derived from the algorithms in *Excel* format.

Keywords: Maximum nodal injection, genetic algorithm, distributed generation

Índice

CAPÍTULO I - INTRODUÇÃO	1
1.1 O PROBLEMA	2
1.2 OBJECTIVOS	4
1.3 ESTRUTURA	5
CAPÍTULO II – ABORDAGEM AO PROBLEMA DA MÁXIMA INJEÇÃO NODAL	6
2.1 FORMULAÇÕES RELACIONADAS COM O PROBLEMA	7
2.2 MOTIVAÇÃO PARA O DESENVOLVIMENTO DE ALGORITMOS	14
CAPÍTULO III – INJEÇÕES NÃO SIMULTÂNEAS	16
3.1 MAXIMA INJEÇÃO NODAL NÃO SIMULTÂNEA.....	17
3.1.1 <i>Máxima Injeção Nodal Não simultânea em Regime ‘n-1’</i>	24
3.1.2 <i>Máxima Injeção Nodal Não simultânea com Localização de Swing Variável</i> 28	
3.1.3 <i>Máxima Injeção Nodal Não simultânea em Regime ‘n-1’ e com Localização de Swing Variável</i>	30
3.2 IMPLEMENTAÇÃO DOS ALGORITMOS	32
3.2.1 <i>Software desenvolvido</i>	32
3.2.2 <i>Trânsito de Energia</i>	36
3.2.3 <i>Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea” à rede de seis barramentos</i>	43
3.2.4 <i>Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea em Regime ‘n-1’” à rede de seis barramentos</i>	48
3.2.5 <i>Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável” à rede de seis barramentos</i>	49
3.2.6 <i>Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea em Regime ‘n-1’ e com Localização de Swing Variável” à rede de seis barramentos</i>	51
3.3 ANÁLISE E COMPARAÇÃO ENTRE OS VÁRIOS ALGORITMOS.....	52
CAPÍTULO IV – INJEÇÕES SIMULTÂNEAS.....	54
4.1 MÉTODO HEURÍSTICO	55
4.2 ALGORITMO GENÉTICO	61
4.3 IMPLEMENTAÇÃO DOS ALGORITMOS	68
4.3.1 <i>Aplicação do algoritmo Heurístico à rede de seis e catorze barramentos</i>	69
4.3.2 <i>Aplicação do algoritmo genético à rede de 6 e 14 barramentos</i>	79
4.4 ANÁLISE E COMPARAÇÃO ENTRE OS VÁRIOS ALGORITMOS.....	88
CAPÍTULO V - CONCLUSÃO	90
5.1 OBSERVAÇÕES FINAIS	91
5.2 PERSPECTIVAS DE TRABALHO FUTURO	93
REFERÊNCIAS BIBLIOGRAFICAS	94
ANEXOS	96

ÍNDICE DE DIAGRAMAS

Diagrama 1 - Descrição do método " Máxima Injeção Nodal Não simultânea" 21	
Diagrama 2 - Descrição do método " Máxima Injeção Nodal Não simultânea em Regime 'n-1'"	27
Diagrama 3 - Descrição do método "Máxima Injeção Nodal Não simultânea com Localização de Swing Variável"	28
Diagrama 4 - Descrição do método "Máxima Injeção Nodal Não simultânea em Regime 'n-1' e com Localização de Swing Variável"	31
Diagrama 5 - Descrição do método Heurístico	60
Diagrama 6 - Descrição do algoritmo Genético	67

ÍNDICE DE FIGURAS

Figura 1 - Rede de 2 barramentos	22
Figura 2 - Rede de 2 barramentos com gerador fictício no barramento 2	22
Figura 3 – Ilustração do método de cálculo para a máxima injeção no nó 2 de uma rede de 3 barramentos em regime N e N-1 de ramos	25
Figura 4 – Ilustração do método de cálculo para a máxima injeção no nó 3 de uma rede de 3 barramentos em regime N e N-1 de ramos	25
Figura 5 – Ilustração da mudança do gerador Swing no decorrer do método “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável” numa rede de 3 barramentos	29
Figura 6 - Esquema do projeto desenvolvido	32
Figura 7 - Plataforma de desenvolvimento Eclipse	33
Figura 8 - Interface gráfico da aplicação desenvolvida	34
Figura 9 - Esquema unifilar de um sistema com n barramentos	36
Figura 10 - Esquema monofásico equivalente de um sistema com n barramentos	37
Figura 11 - Potência em trânsito numa linha genérica k	40
Figura 12 - Rede de 6-barramentos	43
Figura 13 - Rede de 6-barramentos com indicação de ocupação de linha	44
Figura 14 - Resultados após aplicação do algoritmo "Máxima Injeção Nodal Não simultânea " no nó 4	45
Figura 15 - Resultados após aplicação do algoritmo "Máxima Injeção Nodal Não simultânea" no nó 2	46
Figura 16 – Exemplo de codificação de 2 indivíduos	62
Figura 17 – Exemplo da etapa de cruzamento do algoritmo genético.....	63
Figura 18 – Resultado final da etapa de mutação aplicada ao individuo D	64
Figura 19 – Escolha dos melhores indivíduos na etapa Elitismo.....	65
Figura 20 - Interface gráfico da aplicação desenvolvida	68
Figura 21 - Resultados obtidos após aplicação do "Método Heurístico" sem alteração.....	70
Figura 22 – Resultados obtidos após aplicação do "Método Heurístico" com alteração.....	71
Figura 23 – Rede de 14 barramentos sem indicação de taxa de ocupação de linhas	73
Figura 24 - Rede de 14 barramentos com indicação de taxa de ocupação de linhas	74
Figura 25 - Rede com aplicação do método Heurístico utilizando a variante do fator de simultaneidade 1	76
Figura 26 - Rede com aplicação do método Heurístico utilizando a variante do fator de simultaneidade 2	77
Figura 27 - Algoritmo genético aplicado à rede de 6 barramentos tendo em conta a geração existente	80

Figura 28 - Algoritmo genético aplicado à rede de 14 barramentos tendo em conta a geração existente	82
Figura 29 – Algoritmo genético aplicado à rede de 14 barramentos não tendo em consideração a geração existente	84
Figura 30 - Rede de 6-barramentos	97
Figura 31 – Rede de 14 barramentos.....	99

ÍNDICE DE TABELAS

Tabela 1 - Resultados do método "Máxima Injeção Nodal Não simultânea" aplicado à rede de 6 barramentos.....	46
Tabela 2 - Resultados do método "Máxima Injeção Nodal Não simultânea em Regime 'n-1'" quando aplicado à rede de 6 barramentos.....	48
Tabela 3 - Resultados do método "Máxima Injeção Nodal Não simultânea com Localização de Swing Variável" aplicado à rede de 6 barramentos em regime N.....	49
Tabela 4 - Resultados do método "Máxima Injeção Nodal Não simultânea em Regime 'n-1' e com Localização de Swing Variável" aplicado à rede de 6 barramentos	51
Tabela 5 - Potência máxima de injeção individual na rede por nó tendo em os vários métodos.....	52
Tabela 6 – Comparação entre os métodos de Swing variável em regime N e N-1 aplicados à rede de 6 barramentos	53
Tabela 7 - Resultados intermédios do exemplo do método heurístico	57
Tabela 8 - Resultados intermédios do exemplo do método heurístico - após aplicação do fator de simultaneidade	58
Tabela 9 - Resultados finais do exemplo do método heurístico	59
Tabela 10 - Exemplo de uma rede a partir dos dados de um individuo do algoritmo genético	66
Tabela 11 - Resultados obtidos quando aplicado o método Heurístico à rede de 6 barramentos, tendo e não em conta as alterações definidas	71
Tabela 12 - Resultados obtidos quando aplicado o método Heurístico à rede de 14 barramentos, tendo em conta as duas variantes de cálculo do factor de simultaneidade	75
Tabela 13 – Melhor indivíduo aplicado à rede de 6 barramentos tendo em conta a geração presente na rede	79
Tabela 14 - Melhor indivíduo aplicado à rede de 14 barramentos tendo em conta a geração presente na rede.....	81
Tabela 15 - Melhor indivíduo aplicado à rede de 14 barramentos não tendo em consideração a geração presente na rede	83
Tabela 16 - Parâmetros de configuração dos algoritmos genéticos aplicados. 85	
Tabela 17 - Potência máxima de injeção simultânea na rede	88
Tabela 18 - Dados dos Barramentos (geradores e cargas) da rede de 6 barramentos	98
Tabela 19 - Dados das linhas da rede de 6 barramentos.....	98
Tabela 20 - Dados dos Barramentos (geradores e cargas) da rede de 14 barramentos	100
Tabela 21 - Dados das linhas da rede de 14 barramentos.....	101

CAPÍTULO I - INTRODUÇÃO

Apresentação do tema da dissertação e das soluções propostas para o problema subjacente.

1. INTRODUÇÃO

1.1 O PROBLEMA

Os sistemas de energia elétrica têm vindo a assistir a uma mudança de paradigma na forma como têm sido planeados, operados e mantidos. De um sistema em que o planeamento de centros produtores tem como base a evolução de consumo e é caracterizado por centros produtores de grande dimensão em poucos locais, tem-se evoluído para um sistema em que a dimensão média das unidades de geração tendem a diminuir, sendo estas localizadas de forma dispersa nos diferentes níveis de tensão existente.

Neste contexto, os operadores de rede, cuja função, entre outras, é garantir uma elevada qualidade de serviço, deparam-se com um nível de incerteza no que concerne às novas injeções na rede e com um desafio crescente em dimensionar o Sistema de Energia Elétrico. para acomodar estas incertezas.

A introdução em larga escala de geração de energia, a partir de fontes de energia renovável, e as limitações de natureza monetária e ambiental de implementação de reforços de rede têm levado a um aumento na duração de aprovação de projeto, ou mesmo a uma alteração e/ou anulação do mesmo. Consequentemente, todos estes fatores contribuem para uma maior complexidade do problema.

A Geração Distribuída consiste na existência de fontes de geração nos diferentes nós da rede elétrica, permitindo uma gestão do parque produtor que deve ter em consideração a carga na rede e a topologia existente. Em geral, se corretamente operada, pode levar a uma diminuição da sobrecarga das linhas para um crescente e variado conjunto de cenários topológicos e operacionais existentes.

Outro aspeto a ter em conta é o facto de que, com a diminuição da distância entre a geração e o consumo, as perdas de energia provenientes da transmissão vão ser menores.

A correta alocação de geração pelos diferentes nós da rede em fase de planeamento afigura-se muito importante para uma correta exploração do

sistema, permitindo ao operador da rede alcançar uma elevada qualidade de serviço e, ao mesmo tempo, acomodar injeções da rede que podem ser provenientes de diferentes fontes primárias, como sejam, as de origem renovável.

1.2 OBJECTIVOS

Pretende-se com este trabalho estudar, propor e implementar algoritmos que permitam determinar a máxima injeção nodal nos nós de uma rede.

A obtenção dos dados relativos à máxima geração de energia permitida pela rede permite otimizar a rede, identificando os nós ideais para a instalação de novas centrais de geração e minimizando o número de reforços na rede.

Como apresentado no capítulo 2 e 3, as injeções nodais podem ser consideradas como simultâneas ou não simultâneas.

No capítulo 2 são apresentados algoritmos que procedem à determinação da máxima injeção nodal, tendo por base o pressuposto do valor nodal encontrado ser não simultâneo. São ainda identificados os fatores que influenciam este problema e sugeridos valores nodais com base em injeções não simultâneas.

No capítulo 3 são desenvolvidas e propostas soluções que permitem lidar com o problema de injeção simultânea na rede. Em particular, e dada a natureza combinatória do problema, são propostos algoritmos especialmente desenhados e concebidos para resolver este problema.

1.3 ESTRUTURA

A presente dissertação encontra-se dividida em 5 capítulos.

No **segundo capítulo** é realizada uma introdução ao tema e apresentada a sua relevância como objeto de estudo nos últimos anos. São abordadas, de forma resumida, as diferentes metodologias propostas até ao momento e o enquadramento deste projeto nessas metodologias.

No **terceiro capítulo** são desenvolvidos e apresentados algoritmos que permitem a resolução do problema de máxima injeção nodal não simultânea. Neste capítulo são implementados os respetivos algoritmos e analisados os seus resultados.

O **quarto capítulo** consiste na continuação e evolução dos algoritmos apresentados no terceiro capítulo, mas incidido agora nos algoritmos de injeção nodal simultânea. À semelhança do capítulo anterior, neste capítulo são também apresentados e analisados os resultados provenientes das simulações dos algoritmos apresentados neste capítulo.

Por último, no **quinto capítulo** são apresentados os comentários e conclusões finais desta dissertação e indicados possíveis futuros desenvolvimentos para a continuação e melhoramento do trabalho.

CAPÍTULO II – ABORDAGEM AO PROBLEMA DA MÁXIMA INJEÇÃO NODAL

Apresentação dos trabalhos mais relevantes sobre o tema “Capacidade máxima de injeção nodal numa rede”.

2. ABORDAGEM AO PROBLEMA DA MÁXIMA INJEÇÃO NODAL

No levantamento bibliográfico efetuado não foram identificados trabalhos que especificamente abordassem o tema em análise. Contudo, no decurso do presente trabalho foram identificadas algumas publicações que endereçam os temas relacionados com o agora apresentado. Ou seja, que, de uma forma ou de outra, têm como objetivo otimizar as redes de energia elétrica no que concerne à obtenção da capacidade máxima de alocação de energia na rede.

2.1 FORMULAÇÕES RELACIONADAS COM O PROBLEMA

Não sendo objetivo central deste capítulo apresentar um levantamento exaustivo dos trabalhos publicados, apresentam-se de seguida as publicações mais relevantes sobre o tema.

- A metodologia proposta por Harrison e Wallace em [1] pretende endereçar problemas que vão desde o despacho económico à minimização das perdas de energia por transmissão. Esta metodologia consiste em determinar a capacidade máxima de receção de potência nos nós da rede assumindo novos geradores como cargas negativas. Através do OPF (*Optimal Power Flow*), é determinado o despacho das cargas, que representam os novos geradores, maximizando assim a capacidade máxima a instalar.

Este método consiste em associar a cada um dos nós da rede um Custo, C_i . Assim, a maximização da capacidade na rede é conseguida minimizando o custo associado a cada carga, que representa a vantagem em instalar um novo gerador neste ponto. Este permite, igualmente, a análise da influência do aumento de capacidade de produção num determinado nó da rede, sobre a capacidade a instalar nos restantes.

A função objetivo é representada em (2.1) e sujeita às restrições (2.2) a (2.5)

$$\min f(\psi) = \sum_{i=1}^n -C_i \cdot MW_i^0 (1 - \psi_i) \quad (2.1)$$

s.a.:

$$\psi_{MIN,i} \leq \psi_i \leq \psi_{MAX,i} \quad (2.2)$$

$$MVA_i^{DG} = \psi_i \cdot MVA_i^0 \quad (2.3)$$

$$V_{MIN,j} \leq V_j \leq V_{MAX,j} \quad (2.4)$$

$$S_k - S_{MAX,k} \leq 0 \quad (2.5)$$

O fator de ajustamento da capacidade (ψ_i) controla a capacidade de cada um dos geradores que é possível instalar em cada uma das n localizações admitidas. Permite determinar a potência a instalar em função da potência ativa inicialmente admitida, MW^0 , por um valor compreendido entre os limites definidos em (2.2).

O sinal negativo do custo, C_i , representa o benefício na ligação dos novos geradores e assegura um valor máximo negativo para a função objetivo. A restrição (2.3) garante que cada grupo gerador mantém o fator de potência especificado inicialmente.

As restrições (2.4) e (2.5) representam os limites especificados para a tensão em cada barramento j e para o trânsito de potência aparente em cada linha k do sistema em análise.

- Em [2] Harrison e Wallace descrevem a grande importância da Geração Distribuída e como esta tem vindo rapidamente a substituir as grandes gerações centralizadas, devido em parte aos incentivos nacionais e às diretivas da Energia Renovável da União Europeia.

Com a implementação da geração distribuída, os seguintes fatores têm de ser tidos em consideração:

- a possibilidade de ser excedido os limites térmicos dos equipamentos;
- a redução dos níveis de tensão para valores inferiores aos impostos, de forma a garantir a qualidade de serviço;
- o aumento da potência de curto-circuitos e níveis de correntes de curto-circuito;
- a existência de fluxos de potência bidirecionais.

De modo a localizar a máxima capacidade de injeção, é utilizado o método “*Single Bus Injection*” em vários nós da rede. Este método consiste em injetar geração num nó até a rede atingir os seus limites impostos. No capítulo 3.2.1 é apresentada uma metodologia, com base neste método, que tem como objetivo determinar a capacidade máxima que é possível injetar em cada nó da rede. É de salientar que, apesar de este método apresentar a capacidade máxima que é possível injetar em cada nó da rede, não informa, no entanto, sobre a capacidade máxima que é possível injetar na rede.

Este artigo apresenta, assim, um método para o cálculo de capacidade simultânea, que consiste em aumentar a injeção num nó, restringindo os outros sequencialmente, até atingir a máxima potência injetada.

Apesar de este método conseguir, de algum modo, apresentar um aumento na capacidade de potência injetada na rede simultaneamente face à injeção individual, não apresenta, no entanto, a melhor distribuição possível de injeção. Ou seja, é possível alocar mais energia na rede se se tiver em conta a distribuição e local da injeção de nova geração de energia.

- Em [3] Harrison, Piccolo, Siano e Wallace apresentam um método heurístico, os algoritmos genéticos, para o cálculo da capacidade máxima de geração de uma rede. O método procura identificar os locais mais apropriados para ligar os novos grupos geradores e a sua capacidade máxima de geração por local.

São identificadas duas abordagens distintas ao problema:

- identificar o conjunto de localizações mais eficientes para a ligação dos grupos geradores, com base na sua capacidade de produção;
- seleccionar a capacidade mais adequada dos grupos geradores, admitindo ser conhecido o conjunto de pontos de ligação a considerar.

A determinação da localização dos pontos de ligação de um conjunto n de geradores de entre um conjunto m ($m > n$) de possíveis pontos de ligação corresponde a um problema discreto que pode ser resolvido através da utilização de algoritmos genéticos.

De uma forma genérica, nesta primeira abordagem, os algoritmos genéticos são utilizados para determinar o melhor conjunto de pontos de ligação, de modo a que sejam minimizadas as perdas, os custos de ligação e a potência cortada, e traduzindo, assim, uma medida da fiabilidade do sistema.

Relativamente à segunda abordagem, pretende-se otimizar a capacidade dos grupos geradores, admitindo que estes se encontram ligados em locais da rede pré-determinados, e não violando um conjunto de restrições associadas, nomeadamente, às condições de exploração da rede e aos limites de operação de diversos equipamentos.

A maior dificuldade associada a este tipo de abordagens decorre do facto de o utilizador ter de seleccionar um conjunto de locais de entre um elevado número de combinações possíveis de pontos de ligação. Esta pré-selecção afeta desde logo a capacidade total que será possível ligar, pelo que, este requisito constitui uma fragilidade deste tipo de abordagens.

A metodologia descrita nesta publicação combina a resolução de um problema do tipo OPF (Optimal Power Flow – estudo do trânsito de energia de uma rede elétrica) com um algoritmo genético (GA), de modo a pesquisar eficazmente um vasto número de combinações de locais possíveis.

O algoritmo genético apresentado segue as etapas normais – seleção, cruzamento e mutação, criando em cada iteração uma nova população e concluindo quando o critério de convergência é atingido. Este método começa pela identificação do número de grupos geradores que se pretende ligar, a partir do qual é gerada uma população inicial relativa às combinações de pontos de ligação, que são em igual número ao dos geradores. Após a geração da população, os elementos da população, cromossomas, são avaliados através do OPF, o qual pretende identificar a máxima capacidade de geração que é possível ligar considerando estes pontos de ligação.

A função objetivo associada ao algoritmo genético apresenta-se a seguir, no qual a expressão $C_g(P_g)$ representa uma função do tipo quadrático que traduz o benefício que advém da ligação da nova capacidade de produção ao gerador g .

$$f(P_g) = \sum_g C_g(P_g) \quad (2.6)$$

Esta formulação inclui um conjunto de restrições relativas ao equilíbrio de potências ativas (2.7) e reativas (2.8) em cada nó da rede e traduzidas pelas equações AC do trânsito de potências. Assumindo que o t designa as importações/exportações, os índices g e d designam os geradores e cargas, respetivamente, e finalmente o índice k para indicar o nó correspondente. Esta formulação inclui igualmente restrições de limites do módulo da tensão (2.9), valores mínimos e máximos da nova capacidade que se admite poder vir a ligar em cada nó considerado (2.10), limites térmicos dos ramos do sistema traduzidos por restrições relativas ao valor do trânsito de potência aparente em cada ramo (2.11) e, eventualmente, valores limite dos trânsitos de potência em ramos de interligação com outros sistemas adjacentes.

$$\sum_t P_{tk} + \sum_g P_{gk} - \sum_d P_{dk} = 0 \quad (2.7)$$

$$\sum_t Q_{tk} + \sum_g Q_{gk} - \sum_d Q_{dk} = 0 \quad (2.8)$$

$$V_b^{min} < V_b < V_b^{max} \quad (2.9)$$

$$LB_g < P_g < UB_g \quad (2.10)$$

$$S_t < S_t^{max} \quad (2.11)$$

Tal como foi referido anteriormente, o algoritmo genético irá gerar aleatoriamente a população inicial de soluções, correspondendo cada uma delas a uma combinação de nós para a localização dos n grupos geradores. Cada combinação, ou cromossoma, é representada por um vetor de números inteiros, cada um dos quais representa um nó.

Para cada cromossoma é executado um estudo de OPF para determinar a sua aptidão através do cálculo das capacidades que é possível ligar ao conjunto de nós associados a este cromossoma, e não violando as restrições do problema. Após caracterizar todos os elementos de uma população, é criada uma nova população de indivíduos utilizando os mecanismos de seleção já referidos. Este processo iterativo é executado até se verificar uma condição de convergência associada, como por exemplo, a existência de variações do valor da função de avaliação entre populações consecutivas inferior a um valor pré-determinado.

Este método foi implementado em Matlab e testado em redes onde foi admitido um número reduzido de geradores. Os autores chegaram à conclusão que a solução final contém grupos geradores de maior capacidade em nós relativamente afastados entre si eletricamente. No entanto, através dos casos demonstrados, constataram que, no caso de admissão de um número maior de geradores, a capacidade de geração final é maior, mas a geração individual dos geradores é menor.

- Em [4] Karegar, Jalilzadeh, Nabaci e Shabani apresentam o método “*Binary Genetic Algorithm*” de forma a encontrar a melhor solução para o problema. Neste projeto, o problema consiste em encontrar a melhor topologia da geração para uma geração distribuída. Ou seja, distribuir a geração existente pela rede de modo a minimizar os custos totais de compra de energia e as perdas de energia derivadas da transmissão da mesma.

Neste artigo, os autores referem os vários modelos de compra de energia no mercado, como o “*Poolco Model*”, “*Bilateral Contracts Model*”, “*Hybrid Model*”, os seus modos de funcionamento e as vantagens de cada um. Com efeito, é preciso ter em conta que no mercado de energia existem vários vendedores de energia, com diferentes condições e características.

À semelhança do artigo [3], este algoritmo genético também apresenta equações de *fitness* que têm como objetivo otimizar os indivíduos de uma geração para a geração seguinte e limites que restringem os indivíduos para valores aceitáveis.

Os autores apresentam uma técnica interessante na implementação do algoritmo genético, implementada e apresentada no presente trabalho, que consiste no processo adaptativo na etapa de mutação. Durante a etapa de mutação dos genes dos indivíduos da população é executada uma função para otimizar o indivíduo melhorando o gene que vai sofrer a mutação.

Karegar, Jalilzadeh, Nabaci e Shabani concluem que o método proposto permite investigar um conjunto de soluções de modo a encontrar um solução ótima para a topologia da rede em questão, diminuindo o custo total de geração de energia e perda de energia na sua transmissão.

2.2 MOTIVAÇÃO PARA O DESENVOLVIMENTO DE ALGORITMOS

Tal como foi referido no capítulo anterior, pretende-se determinar a capacidade máxima de injeção nodal numa rede, não simultânea e simultaneamente. Deste modo, o desenvolvimento de algoritmos, capazes de obter estes valores de uma forma rápida e objetiva, leva à identificação dos nós ideais da rede para a alocação de novos centros produtores de energia.

De facto, com a introdução crescente de centrais de energia renovável (Eólica, Solar, ...) e conseqüente complexidade na gestão e planeamento das redes, este estudo revela-se, assim, de particular importância na atualidade.

Quais os melhoramentos e contribuições face aos trabalhos referenciados no capítulo anterior?

Nos artigos apresentados anteriormente, é possível observar a investigação de métodos / algoritmos e sua aplicação no estudo das redes de energia elétrica.

Com este trabalho pretende-se desenvolver algoritmos baseados nos artigos acima expostos, de forma a possibilitar a identificação dos passos a seguir e, conseqüentemente, permitir uma melhoria na rede existente, de modo a acondicionar a geração e consumos futuros.

Assim, o objetivo principal deste projeto é determinar as capacidades máximas de injeção de energia permitidas pela rede, a partir das quais se torna possível prever e projetar a rede elétrica para o futuro, tendo em conta as capacidades e limitações da rede em questão. Pois, em certos casos, torna-se mais viável a reorganização da geração na rede, ao invés da construção de novas linhas, de forma a acomodar o aumento da injeção de energia nos locais existentes.

Os projetos referenciados anteriormente tinham como objetivo otimizar a rede através da distribuição da geração existente. Este projeto pretende dar um

passo em frente, identificar a capacidade máxima de injeção nodal que é possível alocar na rede e a sua respectiva distribuição. Assim, perante uma geração já existente na rede, responde à seguinte pergunta: Qual é a potência adicional que é possível injetar na rede e onde?

Este trabalho endereça primeiro as injeções não simultâneas de energia na rede, através de métodos diretos e simples, apresentando, por fim, uma evolução com a apresentação das injeções simultâneas de energia na rede. No decorrer da análise dos resultados, são indicadas as várias limitações e vantagens de cada método, tendo em conta o problema em questão.

CAPÍTULO III – INJEÇÕES NÃO SIMULTÂNEAS

Apresentação dos algoritmos para o cálculo máximo de injeção nodal não simultânea.

3. INJEÇÕES NÃO SIMULTÂNEAS

Os algoritmos presentes neste capítulo procuram resolver o problema da máxima injeção nodal não simultânea. Pretende-se com estes algoritmos implementar uma abordagem simples e intuitiva, com a capacidade de obter resultados sobre um determinado nó específico da rede. São ainda estudadas e analisadas as condições que influenciam o valor determinado da máxima injeção nodal não simultânea na rede, como sejam o caso de considerar o critério 'N-1' e a mudança da localização do gerador Swing.

3.1 MÁXIMA INJEÇÃO NODAL NÃO SIMULTÂNEA

O método “*Máxima Injeção Nodal Não simultânea*” é um método direto de obter a injeção máxima de energia que se pode ter num determinado nó, tendo em consideração as limitações dos ramos da rede e as características técnicas do gerador que faz o balanço (gerador Swing).

Função objetivo para o nó k:

$$f(P_{G_k}) = \max \{P_{G_k}\} \quad (3.1)$$

com,

$$k = 1, \dots, n$$

$$P_G = P_{G_1}, \dots, P_{G_k}$$

s.a.

$$S'_{Linha} < S_{Linha} \quad (3.2)$$

$$P_{G=2} > 0 \quad (3.3)$$

As restrições (3.2) representam o facto de as linhas não poderem ter uma ocupação superior a 100% do seu rate e do gerador *Swing* não poder ter uma geração inferior a zero (3.3), limite técnico mínimo que se assumiu como $P_{min} = 0$.

Quanto ao limite máximo das linhas, este parâmetro pode ser configurável. Ou seja, no caso de se pretender ter alguma flexibilidade na rede pode-se configurar o limite superior de ocupação das linhas para, por exemplo 80%, garantindo alguma segurança e flexibilidade na rede. Ao longo deste trabalho, o limite superior de ocupação das linhas, sendo estas elementos limitadores, foi fixado nos 100%, uma vez que o objetivo é obter a capacidade máxima de injeção de energia na rede.

Para alcançar o resultado final, este método usa a metodologia baseada no algoritmo “*Quicksort*”, não para ordenação de valores (função para o qual este método foi desenvolvido), mas para a procura do valor ideal, garantindo uma rápida convergência.

Em cada etapa o sistema é testado, usando o PSSE para verificar se a solução é válida mediante as restrições impostas.

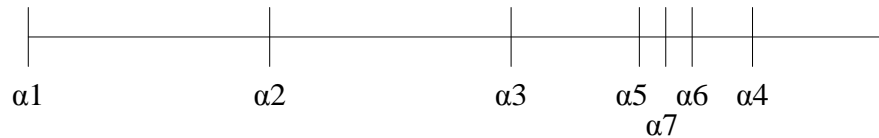
O que é o algoritmo “*Quicksort*” ? Como funciona o método “*Máxima Injeção Nodal Não simultânea*” usando o algoritmo modificado “*Quicksort*”?

O algoritmo *Quicksort* é um método de ordenação rápido e eficiente inventado por C. A. R. Hoare em 1960. [16]

Neste algoritmo é seguido a metodologia do *Quicksort*, onde é definido um determinado valor como potência de injeção, o qual vai servir como passo (intervalo). Começa-se por injetar no respetivo nó o passo e vamos sempre incrementando a injeção com o passo até uma das restrições ser atingida. Sendo o valor do passo anterior válido, divide-se o passo e incrementa-se ao último valor válido. Segue-se este procedimento até o passo usado ser igual ou inferior à precisão especificada.

Exemplo:

Determinar a capacidade máxima de potência que é possível injetar no nó λ da rede.



Começa-se por injetar uma potência definida na rede, potência esta cujo valor representa o passo, ΔP_k , e de seguida a rede é validada,

$$\alpha_2 = \alpha_1 + \Delta P_k \Rightarrow \alpha_2 \text{ é válido}$$

$$\alpha_3 = \alpha_2 + \Delta P_k \Rightarrow \alpha_3 \text{ é válido}$$

$$\alpha_4 = \alpha_3 + \Delta P_k \Rightarrow \alpha_4 \text{ não é válido}$$

Ao injetar a potência α_4 a rede não é válida de acordo com os critérios impostos. Logo, o passo é reduzido para metade do intervalo entre o valor válido, α_3 , e o não válido, α_4 e é acrescentado ao último valor válido, α_3 . Assim, fica:

$$\Delta P_k' = \frac{\alpha_4 - \alpha_3}{2}$$

$$\alpha_5 = \alpha_3 + \Delta P_k' > \alpha_5 \text{ é válido}$$

Volta-se a definir o valor do passo como metade do intervalo entre o último valor válido e o não válido e acrescenta-se ao último valor válido.

$$\Delta P_k'' = \frac{\alpha_4 - \alpha_5}{2}$$

$$\alpha_6 = \alpha_5 + \Delta P_k'' \Rightarrow \alpha_6 \text{ **não é válido**}$$

$$\Delta P_k''' = \frac{\alpha_5 - \alpha_6}{2}$$

$$\alpha_7 = \alpha_5 + \Delta P_k''' \Rightarrow \alpha_7 \text{ **é válido**}$$

Como,

$$\alpha_6 - \alpha_7 < \Delta x$$

Ou seja, como o intervalo final é menor que a precisão pretendida, Δx , α_7 é a solução final. Concluindo, para este exemplo, o máximo de potência que é possível injetar no nó λ é α_7 .

De seguida, é apresentado um diagrama de blocos que demonstra o procedimento deste método descrito em cima.

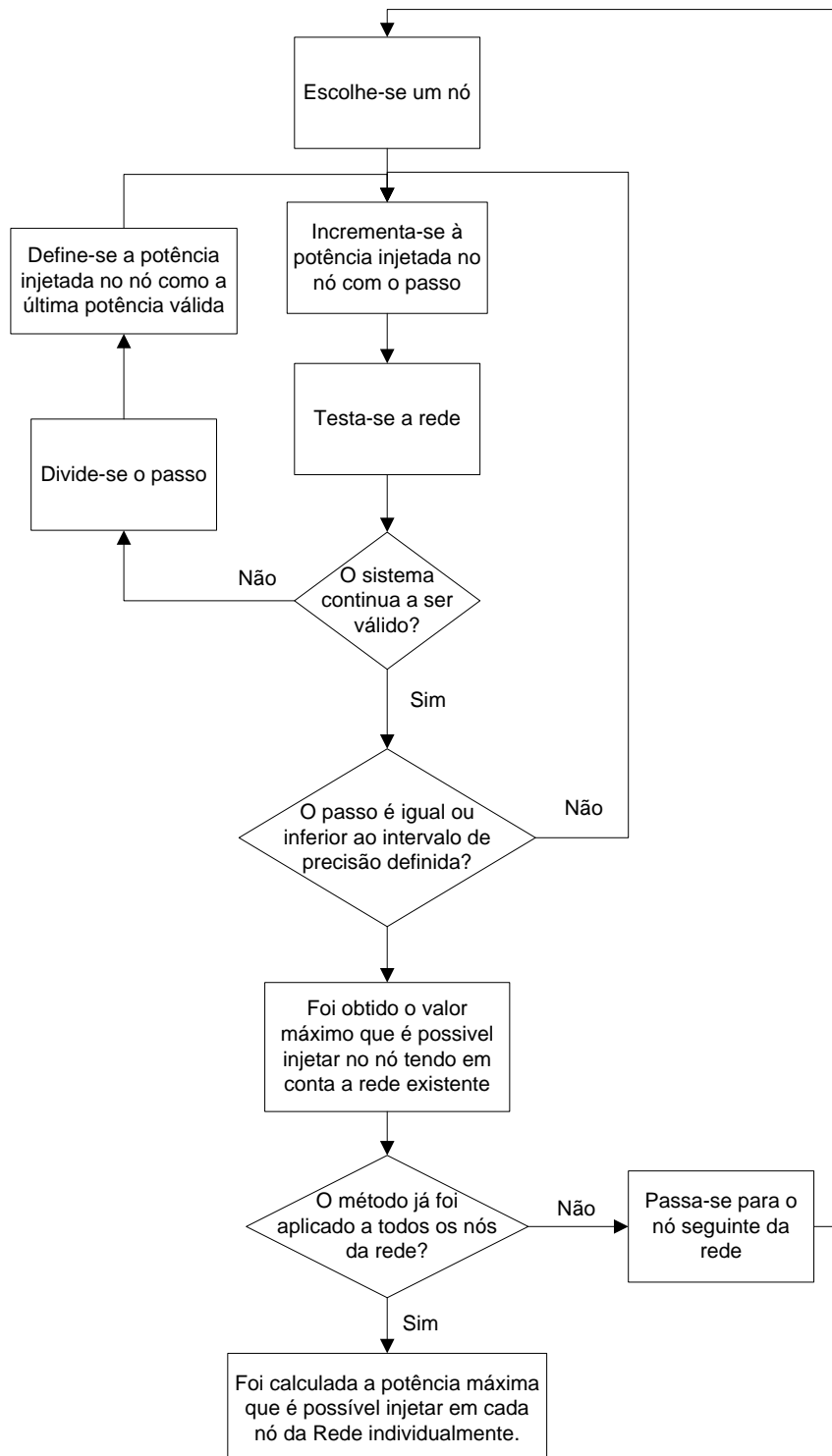


Diagrama 1 - Descrição do método " Máxima Injeção Nodal Não simultânea"

A seguir, é apresentado um modelo simplificado com dois nós, de modo a demonstrar este método.

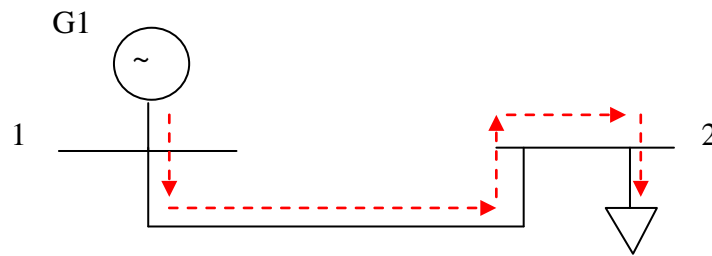


Figura 1 - Rede de 2 barramentos

Como é possível observar no esquema apresentado em cima, a energia flui do gerador do nó 1 para o nó 2 onde está a carga, passando pela linha que une os dois nós.

Este método consiste em criar um gerador em todos os nós sequencialmente (exceto onde existe um gerador do tipo Swing) e injetar a energia. Como esta rede só tem dois nós e um deles tem um gerador Swing, só se pode criar um gerador no nó 2.



Figura 2 - Rede de 2 barramentos com gerador fictício no barramento 2

Como está representado na Figura 2, insere-se um gerador no nó 2 e começa-se a injetar energia pelo método descrito em cima até se determinar a potência máxima, tendo em conta as restrições impostas. Como é possível observar pelo esquema, após a inserção do novo gerador no nó 2, facilmente se conclui que a energia fornecida por este gerador é consumida pela carga que também se encontra neste nó. A potência gerada pelo gerador 2 corresponde, assim, à

potência necessária para a carga e é igual à potência inicial gerada pelo gerador 1 menos as perdas derivadas da transmissão de energia, uma vez que a energia agora não necessita de passar pela linha. De notar que esta rede é um modelo simplificado e que à medida que a rede é maior também o é a complexidade do problema e da solução. Uma breve referência e exemplo disso é o facto de uma rede não depender apenas de um gerador *Swing*, mas também de vários geradores com potência gerada constante.

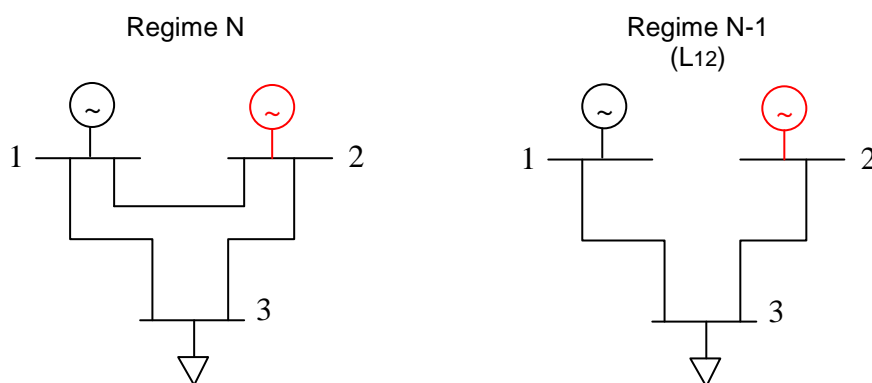
3.1.1 Máxima Injeção Nodal Não simultânea em Regime 'n-1'

O presente método tem como objetivo estudar as diferenças da máxima capacidade de injeção numa rede em regime N-1 e suas influências no funcionamento da rede.

Definição de Regime de contingência 'n-1' - Considera-se a falha de um qualquer elemento da RNT (linha simples, circuito de linha dupla, grupo gerador, autotransformador, bateria de condensadores), devendo nos restantes, sem exceção, não se verificarem violações dos critérios de tensão e de sobrecarga, sem qualquer redespacho ou reconfiguração topológica. [17]

Este algoritmo segue a mesma metodologia que o método “Máxima Injeção Nodal Não simultânea”, mas cada vez que a rede é validada verifica-se se a rede também é válida para o regime N-1. O regime N-1 estudado neste trabalho e apresentado ao longo deste subcapítulo apenas inclui as contingências não simultâneas de um elemento da rede, neste caso o único elemento estudado são as linhas simples. Para se simular este regime coloca-se sequencialmente todas as linhas, uma a uma, fora de serviço e verifica-se se a rede continua válida de acordo com as restrições/limitações impostas.

Os esquemas seguintes tentam demonstrar este método.



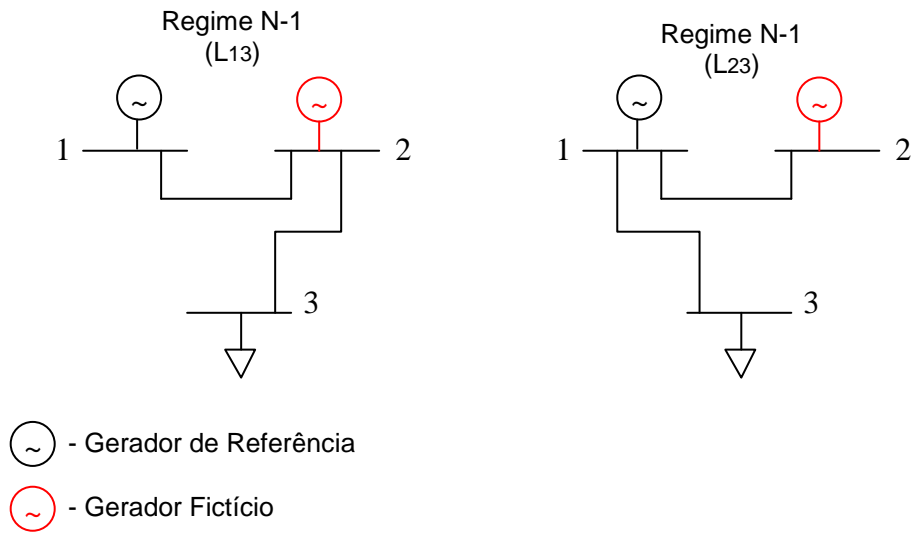


Figura 3 – Ilustração do método de cálculo para a máxima injeção no nó 2 de uma rede de 3 barramentos em regime N e N-1 de ramos

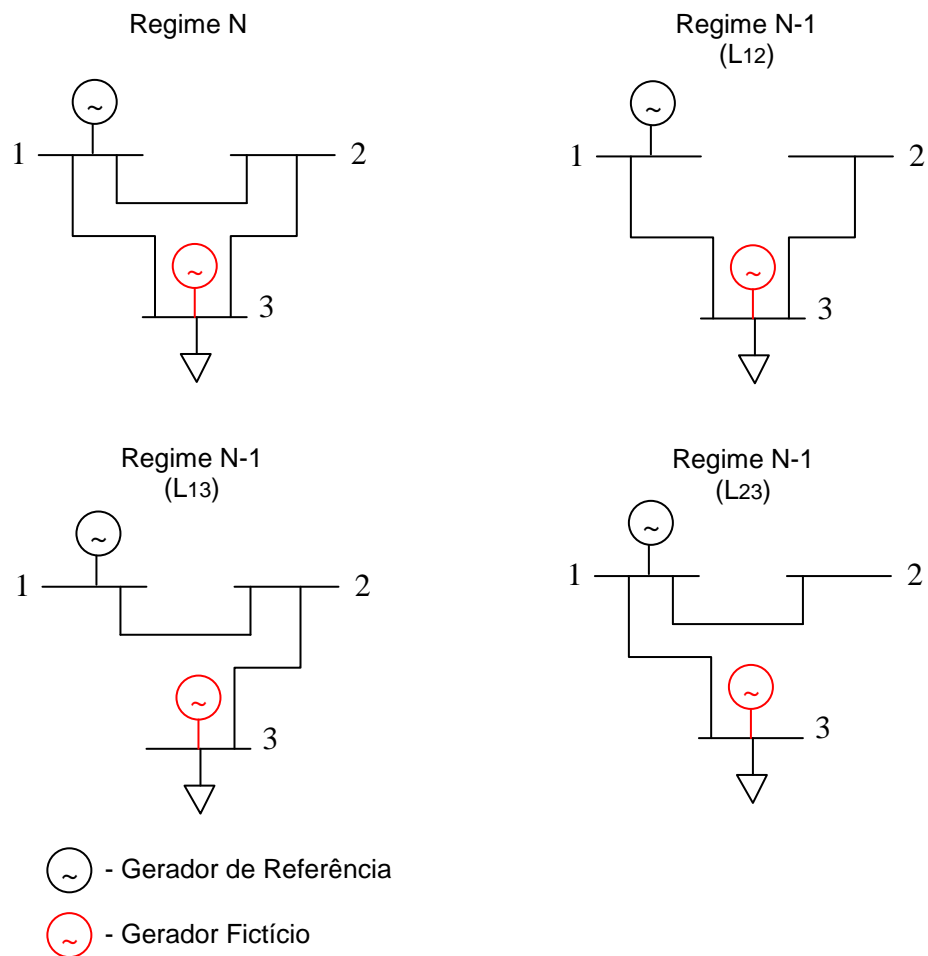


Figura 4 – Ilustração do método de cálculo para a máxima injeção no nó 3 de uma rede de 3 barramentos em regime N e N-1 de ramos

Como ilustrado na Figura 3 e na Figura 4, a potência máxima que é possível injetar no nó 2 calcula-se quando a linha que liga o nó 1 ao nó 2 está fora de serviço, depois quando a linha 1-3 está fora de serviço e por fim a linha 2-3. Após o cálculo para estes três casos, a potência máxima que é possível injetar no nó 2 corresponde ao menor valor calculado em cada uma destas contingências.

Para o nó 3 procede-se da mesma maneira, colocando cada uma das linhas fora de serviço sequencialmente. No entanto, não se calcula para o nó 1, uma vez que o gerador neste nó já é o gerador Swing.

Tendo em conta o regime N-1, este algoritmo permite saber qual a potência máxima que é possível injetar numa rede num determinado nó, tendo em conta que qualquer uma das linhas pode sair fora de serviço.

$$\Delta P_k < \Delta x$$

O diagrama de blocos seguinte demonstra o funcionamento deste método.

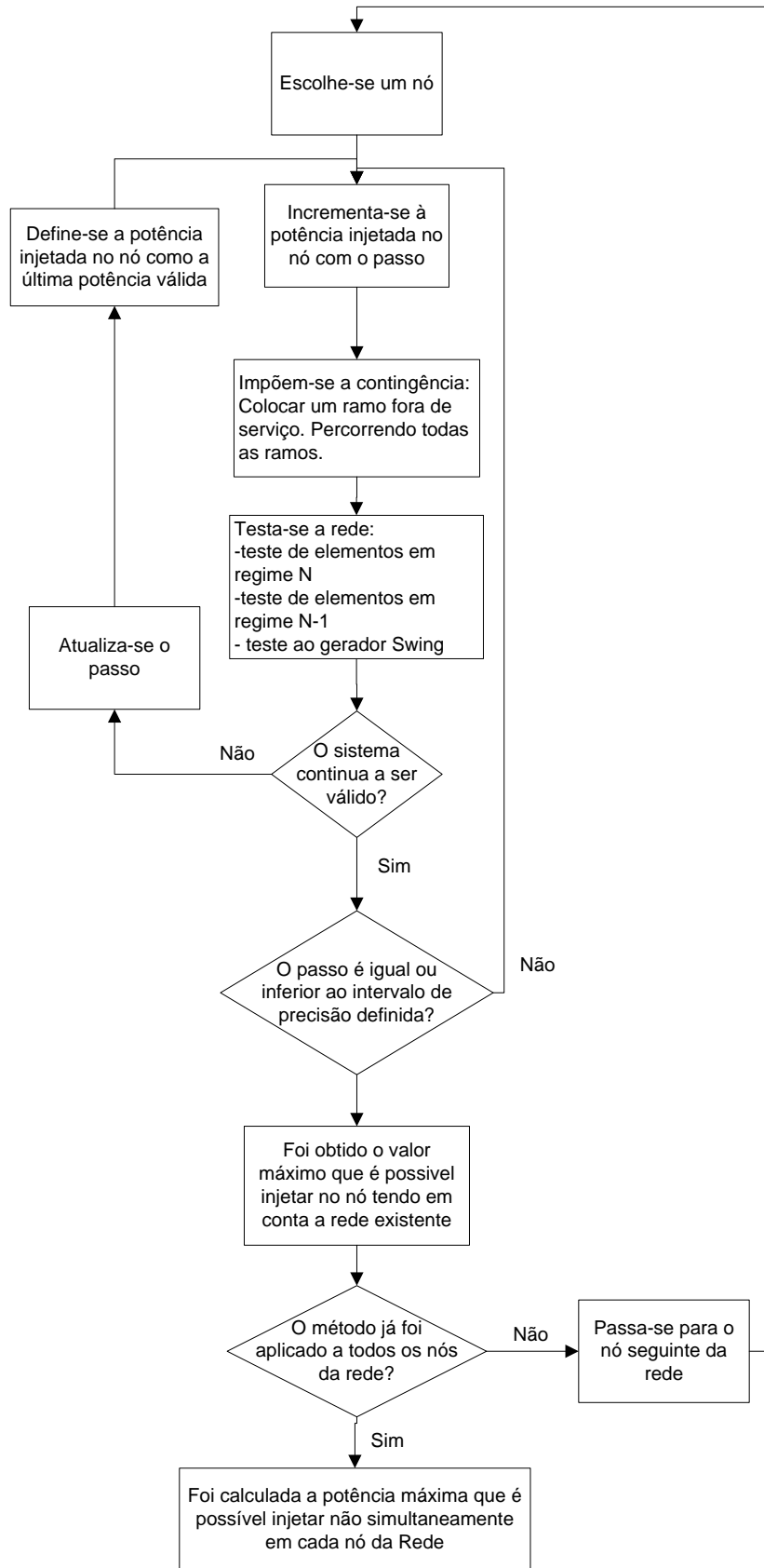


Diagrama 2 - Descrição do método " Máxima Injeção Nodal Não simultânea em Regime 'n-1'"

3.1.2 Máxima Injeção Nodal Não simultânea com Localização de Swing Variável

Com o seguinte método, é calculada a potência máxima que é possível injetar na rede, com a localização do gerador Swing a variar entre os geradores presentes na rede.

O método de cálculo é em tudo semelhante ao método “Máxima Injeção Nodal Não simultânea” apresentado no subcapítulo anterior, mas com o funcionamento como gerador Swing a percorrer todos os geradores já presentes na rede.

Ou seja, para o caso da rede apresentada no subcapítulo anterior (ver Figura 5), calcula-se o máximo de potência que é possível injetar em cada barramento, considerando o gerador 1 como o gerador *Swing*. De seguida, considera-se o gerador 2 o gerador *Swing* e assim, sucessivamente, até todos os geradores terem sido o gerador *Swing*, calculando para cada caso a potência máxima que é possível injetar em cada barramento.

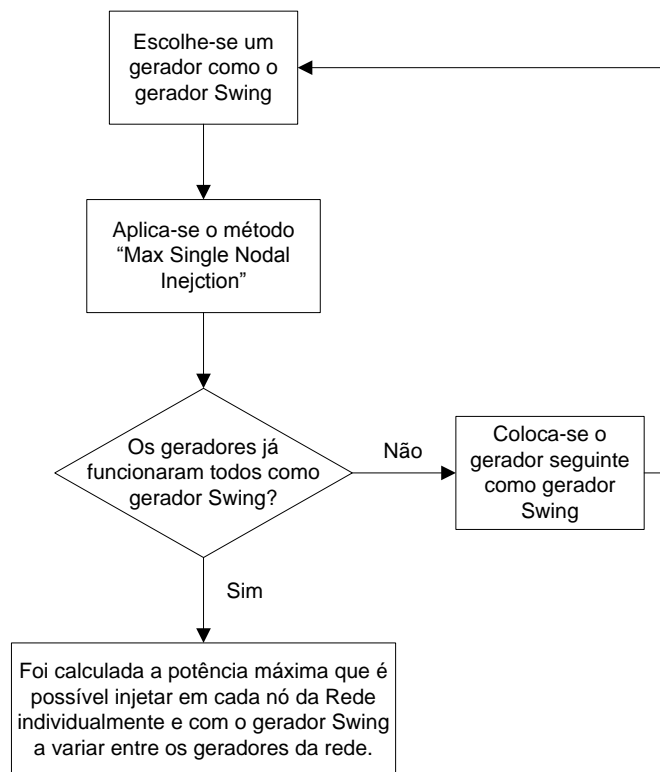
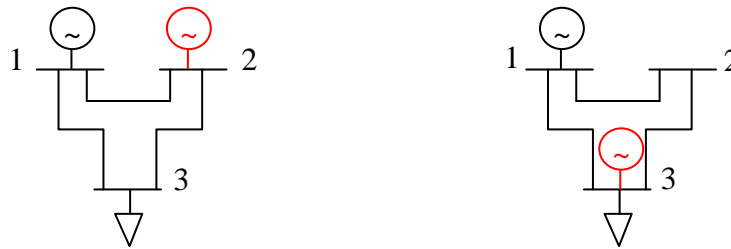


Diagrama 3 - Descrição do método "Máxima Injeção Nodal Não simultânea com Localização de Swing Variável"

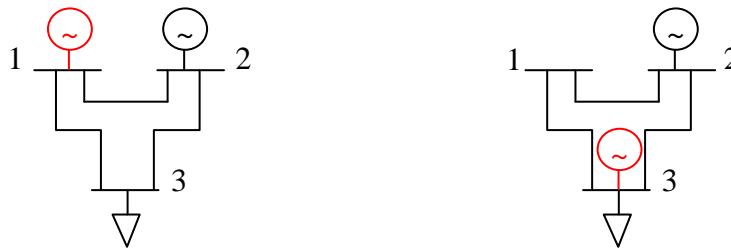
Ao alterar a função de gerador Swing de um gerador para outro, a rede vai-se alterar, pois este gerador vai tentar compensar a rede.

As figuras seguintes representam uma rede de 3 barramentos com um gerador Swing e a sua mudança de localização no decorrer da execução do método.

Gerador Swing no 1º barramento



Gerador Swing no 2º barramento



Gerador Swing no 3º barramento





-  - Gerador Swing
-  - Gerador Fictício

Figura 5 – Ilustração da mudança do gerador Swing no decorrer do método “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável” numa rede de 3 barramentos

3.1.3 Máxima Injeção Nodal Não simultânea em Regime 'n-1' e com Localização de Swing Variável

O algoritmo apresentado neste subcapítulo é uma combinação dos métodos apresentados em 3.1.1 e 3.1.2. Pretende-se o estudo da capacidade máxima de potência que é possível injetar na rede, tendo em conta todas as possibilidades definidas. Ou seja, com o gerador Swing a variar entre os vários geradores, testando para cada caso o regime N-1.

O método de cálculo é semelhante ao método “Máxima Injeção Nodal Não simultânea em Regime 'n-1'” apresentado anteriormente. Porém, este apresenta uma etapa adicional, que consiste em alternar o funcionamento como gerador Swing entre todos os geradores já presentes na rede, à semelhança do funcionamento do método “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável”.

Para o caso de uma rede de 3 barramentos, calcula-se o máximo de potência que é possível injetar em cada barramento. Inicialmente, considera-se o gerador 1 como o gerador *Swing*, tendo em conta o regime N-1. De seguida, considera-se o gerador 2 o gerador Swing e, à semelhança da etapa anterior, calcula-se a máxima potência que é possível injetar em cada barramento, tendo em conta o regime N-1. E assim, sucessivamente, até todos os geradores presentes na rede terem funcionado como gerador Swing.

O diagrama seguinte ilustra o funcionamento deste método, o qual é em tudo semelhante ao método “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável” com a diferença de, neste método, se ter de validar a rede para o regime N-1.

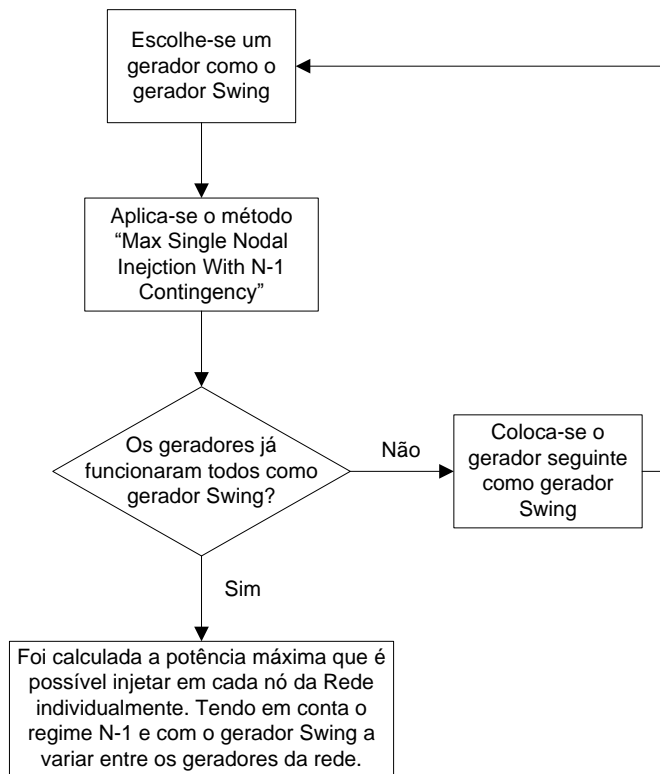


Diagrama 4 - Descrição do método "Máxima Injeção Nodal Não simultânea em Regime 'n-1' e com Localização de Swing Variável"

3.2 IMPLEMENTAÇÃO DOS ALGORITMOS

3.2.1 Software desenvolvido

A arquitetura concebida para a implementação do projeto está esquematizada no seguinte diagrama:

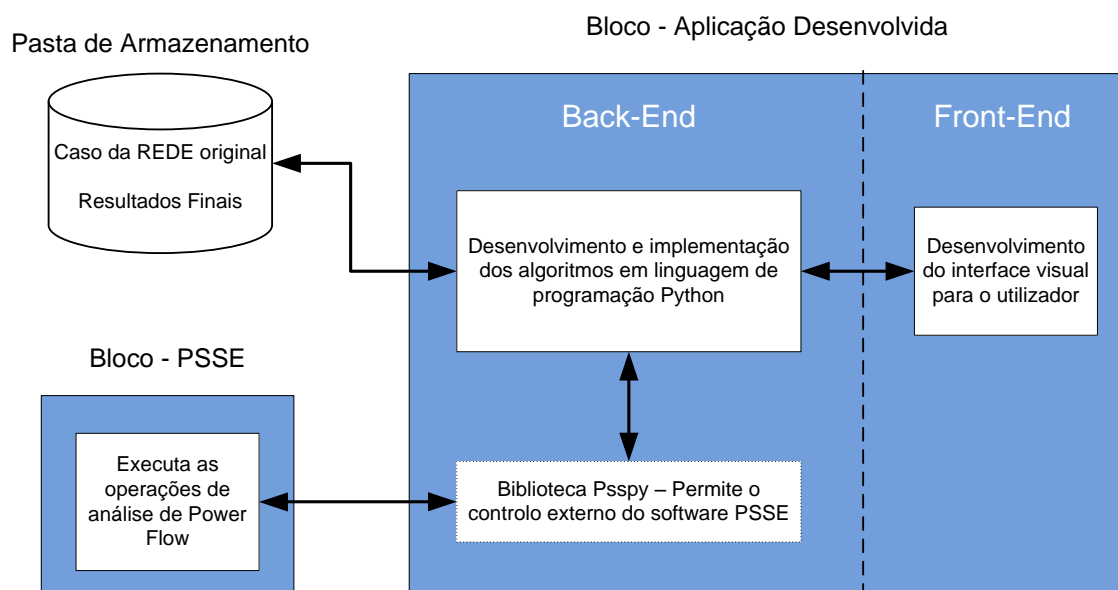


Figura 6 - Esquema do projeto desenvolvido

Na Figura 6 é possível observar claramente 2 blocos funcionais:

O primeiro bloco representa a Aplicação Desenvolvida. Este bloco contém dois módulos, o *Back-End* e o *Front-End*.

O módulo *Back-End* foi desenvolvido na linguagem de programação *Python* [10] com recurso à plataforma de desenvolvimento *Eclipse* [11], ver Figura 7. Este é o módulo principal da aplicação, pois é aqui que estão implementados todos os algoritmos e funções necessárias ao seu funcionamento. Contém a implementação dos algoritmos e de todas as funções necessárias à execução dos mesmos.

No Anexo B é possível observar parte do código implementado no módulo Back-End.

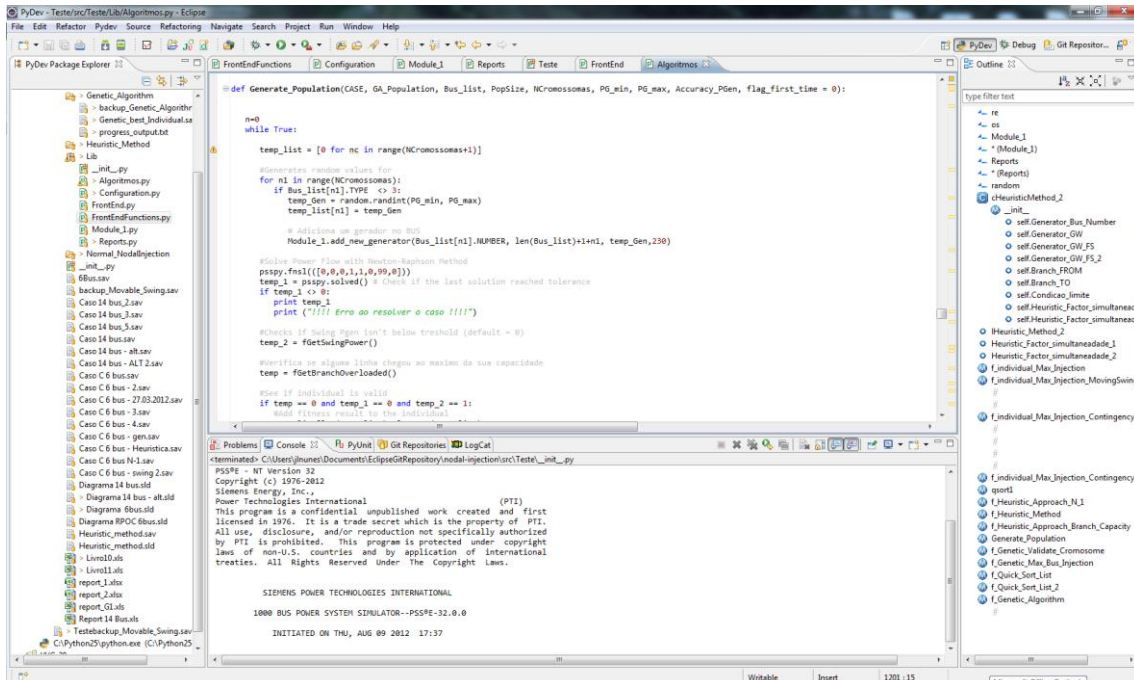


Figura 7 - Plataforma de desenvolvimento Eclipse

O módulo *Front-End* nasceu da necessidade de permitir um interface gráfico e de fácil uso para o operador, onde este pode executar os algoritmos desejados e definir os seus parâmetros, ver Figura 8. Para além de permitir, através de comandos, a execução dos algoritmos, permite igualmente a possibilidade de criar relatórios com os resultados dos algoritmos executados. Este módulo foi implementado através da linguagem de programação Python, fazendo uso das bibliotecas WxPython [14] e Wxglade [13], que servem para criar objetos visuais.

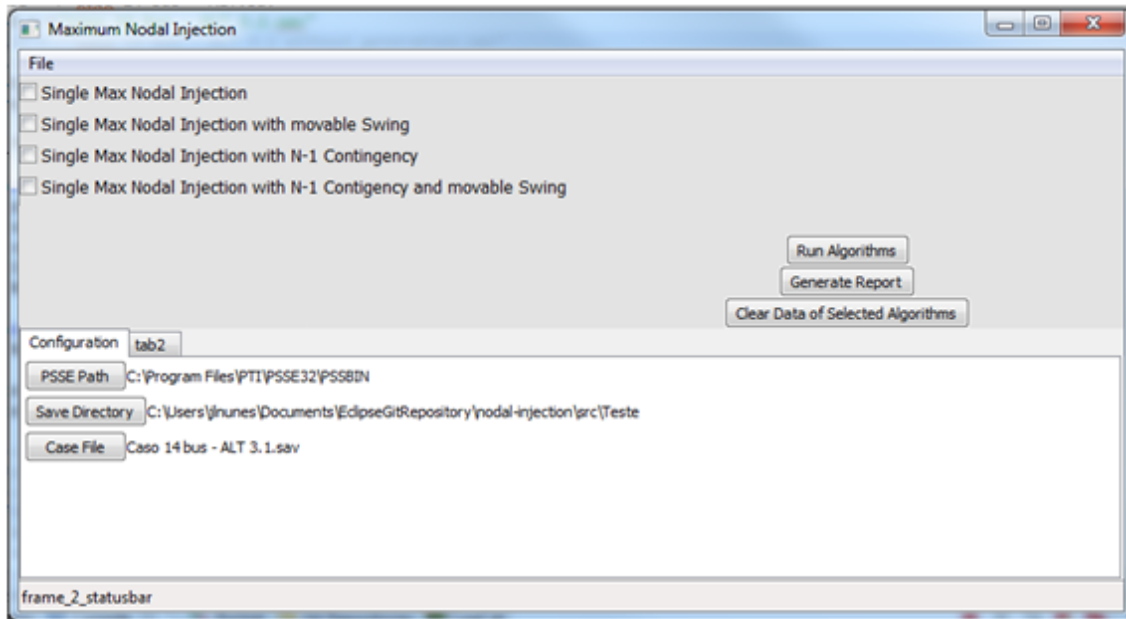


Figura 8 - Interface gráfico da aplicação desenvolvida

O segundo bloco representa o software da Siemens PSSE e é neste software onde vão ser executadas as funções para analisar o trânsito de energia da rede e alterar os elementos da mesma. O controlo externo desta aplicação só é possível através da biblioteca “*psspy*”, a qual está embebida no módulo Back-End. Ponto importante, uma vez que foi o facto de a biblioteca estar desenvolvida na linguagem de programação Python que levou ao desenvolvimento da aplicação em Python, caso contrário podia-se ter escolhido outra linguagem de programação (C, C++, C#, java, Matlab, ...).

No interface da aplicação é possível escolher o método que se pretende executar, bem como o caso que se pretende utilizar (ver Figura 8). Após a execução do método, a aplicação gera uma pasta que contém o caso da rede final, bem como todos os relatórios associados ao programa PSSE aquando a execução do método. Os resultados finais podem ser exportados para formato Excel através do botão “Generate Report” (ver Figura 8). Todos os ficheiros gerados pela aplicação são gravados na pasta de armazenamento (ver Figura 6).

É de salientar que, no decorrer da implementação da aplicação, foi desenvolvido um conjunto de funções e estruturas necessárias à execução dos algoritmos, capazes de armazenar os dados das redes, executar os comandos de *Power Flow*, bem como alterar os dados dos elementos das redes.

Por exemplo, no método “Máxima Injeção Nodal Não simultânea”, o algoritmo tem de criar um gerador fictício no nó que está a estudar. Para isso, foram criadas funções para lerem o ficheiro da rede, alterar a rede e voltar a gravar a rede para a execução do método. Cada vez que o valor do nó é incrementado, o valor do gerador fictício tem de ser alterado, ou seja, tem de se ler a rede, alterar o valor do gerador fictício e voltar a gravar a rede, para se poder executar os comandos de trânsito de energia pelo PSS/E. No final deste algoritmo, cria-se uma pasta que contém o caso da rede original, para efeitos de comparação e backup, um caso correspondente a cada nó, pois o algoritmo cada vez que é executado verifica a potência máxima que é possível injetar em cada nó. Logo, tem-se um número de resultados igual ao número de nós da rede (menos o nó do Swing), o que origina o mesmo número de casos de redes. A pasta contém ainda um log com informação resultante das operações executadas pelo PSS/E.

O ficheiro Excel com o relatório dos resultados é criado à parte, mediante o comando do operador. Deste modo, é possível selecionar vários métodos e obter um ficheiro Excel com o relatório dos resultados correspondentes aos algoritmos selecionados.

3.2.2 Trânsito de Energia

O Trânsito de Energia, também conhecido como Trânsito de Potência ou Fluxo de Potência (Power Flow, Load Flow), corresponde à solução, em regime estacionário, de um sistema de energia elétrica, compreendendo os elementos que o compõem, nomeadamente geradores, cargas e a própria rede.

O Trânsito de Energia consiste no cálculo das amplitudes e argumentos das tensões de todos os barramentos (nós) da rede e das potências ativas e reativas que transitam na rede, para condições de geração e carga especificadas e para uma dada configuração topológica da rede.

O cálculo do trânsito de energia numa rede resume-se nos seguintes passos:

- Formulação do modelo matemático representativo do sistema;
 - Especificação do tipo de barramento;
 - Obtenção de soluções numéricas das equações do trânsito de energia, as quais fornecem o valor das amplitudes e argumentos das tensões em todos os barramentos;
 - Cálculo das potências transitadas nos ramos.
-
- Para a obtenção do modelo matemático considera-se a rede da figura seguinte.

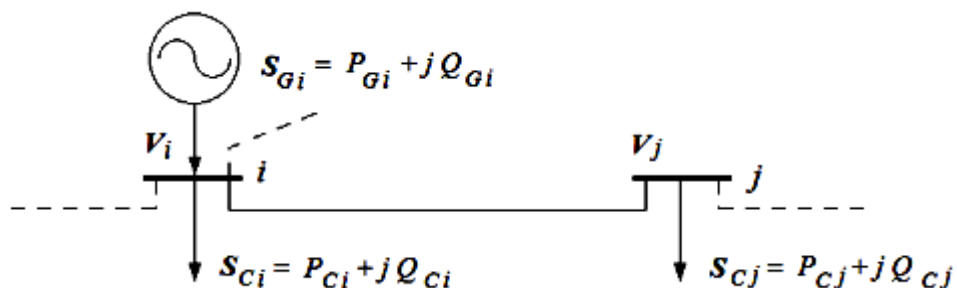


Figura 9 - Esquema unifilar de um sistema com n barramentos

Considerando o barramento genérico i do sistema representado na Figura 9, ao qual se encontram ligados um gerador G_i , uma carga C_i e uma linha de transmissão k , a potência injetada vem:

$$S_i = S_{Gi} - S_{Ci} = P_{Gi} - P_{Ci} + j(Q_{Gi} - Q_{Ci}) \quad (3.4)$$

A linha de transmissão k ligada entre os nós i e j e representada na Figura 10, está representada no modelo equivalente em π , caracterizado por uma impedância longitudinal Z_L e uma admitância transversal $\frac{Y_T}{2}$ em cada extremo da linha.

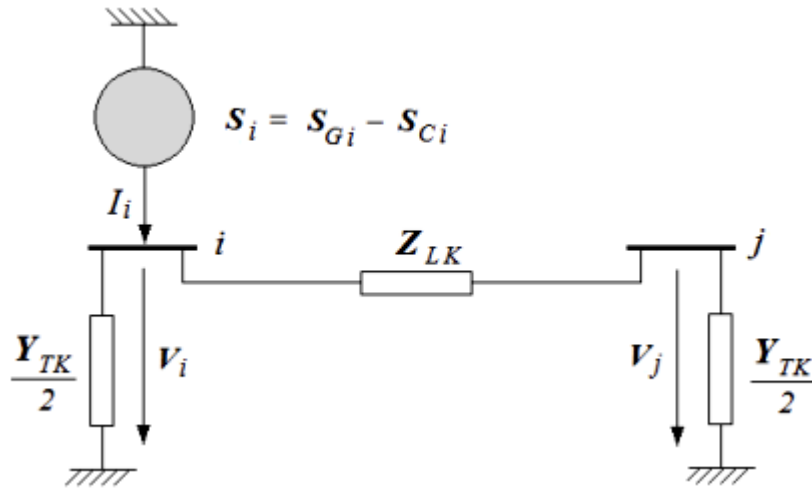


Figura 10 - Esquema monofásico equivalente de um sistema com n barramentos

Aplicando a primeira lei de Kirchoff ao barramento i tem-se a seguinte equação:

$$\frac{S_i^*}{V_i^*} = \sum_{j=1}^n \sum_{j \neq i} \frac{V_{TK}}{2} V_i + \sum_{j=1}^n \sum_{j \neq i} \frac{1}{Z_{Lk}} (V_i - V_j) = \sum_{j=1}^n \sum_{j \neq i} \left(\frac{V_{TK}}{2} + \frac{1}{Z_{Lk}} \right) V_i + \sum_{j=1}^n \sum_{j \neq i} \left(-\frac{1}{Z_{Lk}} \right) V_j \quad (3.5)$$

Onde,

$$\frac{S_i^*}{V_i^*} = y_{ii} V_i + \sum_{j=1}^n \sum_{j \neq i} y_{ij} V_j = \sum_{j=1}^n y_{ij} V_j \quad (3.6)$$

Da equação anteriormente obtém-se:

$$P_i - jQ_i = V_i^* \sum_{j=1}^n y_{ij} V_j, \quad i = 1, \dots, n \quad (3.7)$$

Ou substituindo a equação (3.4) em (3.7),

$$P_{Gi} - P_{Ci} + j(Q_{Gi} - Q_{Ci}) = V_i^* \sum_{j=1}^n y_{ij} V_j, \quad i = 1, \dots, n \quad (3.8)$$

A equação (3.8) representa a forma complexa das equações de trânsito de energia.

Tendo em conta a tensão complexa em notação polar,

$$V_i = V_i e^{j\theta_i} \quad (3.9)$$

e a admitância complexa em notação retangular,

$$y_{ij} = G_{ij} + jB_{ij} \quad (3.10)$$

onde G_{ij} e B_{ij} são a condutância e a susceptância do ramo i, j , respetivamente.

Substituindo estas equações na equação 3.8, obtém-se as equações reais, em coordenadas polares das tensões, que exprimem o equilíbrio de potência ativa e reativa injetada no barramento i ,

$$P_i = P_{Gi} - P_{Ci} = \sum_{j=1}^n V_i V_j [G_{ij} \cos(\theta_i - \theta_j) + B_{ij} \sin(\theta_i - \theta_j)], \quad i = 1, \dots, n \quad (3.11)$$

$$Q_i = Q_{Gi} - Q_{Ci} = \sum_{j=1}^n V_i V_j [G_{ij} \sin(\theta_i - \theta_j) - B_{ij} \cos(\theta_i - \theta_j)], \quad i = 1, \dots, n \quad (3.12)$$

Decompondo as tensões nodais em parte real e imaginária,

$$V_i = V_{Real_i} + jV_{Imag_i} \quad (3.13)$$

Obtemos as equações do trânsito de energia, sob a forma real, em coordenadas retangulares.

$$P_i = P_{Gi} - P_{Ci} = \sum_{j=1}^n \left[G_{ij} \left(V_{Real_i} V_{Real_j} + V_{Imag_i} V_{Imag_j} \right) + B_{ij} \left(V_{Real_j} V_{Imag_i} - V_{Real_i} V_{Imag_j} \right) \right], \quad i = 1, \dots, n \quad (3.14)$$

$$Q_i = Q_{Gi} - Q_{Ci} = \sum_{j=1}^n \left[G_{ij} \left(V_{Real_j} V_{Imag_i} - V_{Real_i} V_{Imag_j} \right) + B_{ij} \left(V_{Real_i} V_{Real_j} + V_{Imag_i} V_{Imag_j} \right) \right], \quad i = 1, \dots, n \quad (3.15)$$

- Nos sistemas de energia elétrica consideram-se três tipos de barramentos, *PQ*, *PV* e *Balanço*, em função das variáveis conhecidas nesses mesmos barramentos.

Os barramentos tipo PQ (geração ou carga) são os mais comuns nas redes elétricas e correspondem, em geral, a barramentos de carga, nos quais se conhecem as potências ativa e reativa. Calcula-se as tensões nestes barramentos, em amplitude e argumento, as quais dependem das potências geradas noutros barramentos e das perdas no sistema.

Os barramentos tipo PV (geração) correspondem aos barramentos de geração, nos quais é imposta a potência ativa a ser gerada e o valor da amplitude da tensão. A potência reativa, gerada nestes barramentos, bem como o argumento da tensão, são calculados de modo a garantir o valor da amplitude da tensão imposto. Nestes barramentos, são conhecidas as potências ativas e reativas das cargas.

Os barramentos do tipo *Balanço*, denominado ao longo do trabalho como *Swing*, têm como objetivo garantir o equilíbrio de potências geradas com as potências de carga e perdas do sistema. Num sistema de energia elétrica existe pelo menos um nó de balanço que é também utilizado como referência

para a tensão, dado que a amplitude e o argumento desta são impostos neste tipo de barramento. As potências ativa e reativa são calculadas.

Após definidas as equações (3.14) e (3.15), a solução do trânsito de energia consiste em determinar as tensões nos barramentos, a potência injetada no nó de balanço e as potências transitadas nas linhas:

- No cálculo da amplitude e argumento das tensões nos barramentos, uma vez que as equações do trânsito de energia são não-lineares, utiliza-se o método iterativo Newton-Raphson [7].
- O passo seguinte consiste no cálculo da potência injetada no barramento de balanço através da equação (3.7).
- As potências ativas e reativas nas linhas são calculadas através do cálculo da potência complexa, S_{ij}^k , que transita na linha L_{ik} , e medida junto ao nó i e j . Ver Figura 11.

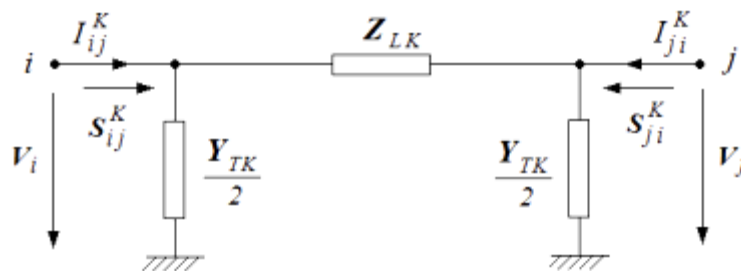


Figura 11 - Potência em trânsito numa linha genérica k

Potência complexa que transita na linha k do nó i para o j ,

$$S_{ij}^k = P_{ij}^k + jQ_{ij}^k = V_i(I_{ij}^k)^* = \frac{1}{Z_{Lk}^*} (V_i^2 - V_i V_j^*) + \frac{Y_{Tk}^*}{2} V_i^2 = \left(\frac{1}{Z_{Lk}^*} + \frac{Y_{Tk}^*}{2} \right) V_i^2 - \frac{1}{Z_{Lk}^*} V_i V_j^* \quad (3.16)$$

e na linha que transita do nó j para o i ,

$$S_{ji}^k = P_{ji}^k + jQ_{ji}^k = V_j (I_{ji}^k)^* = \left(\frac{1}{Z_{Lk}^*} + \frac{Y_{Tk}^*}{2} \right) V_j^2 - \frac{1}{Z_{Lk}^*} V_i V_j^* \quad (3.17)$$

Onde,

$$Z_{Lk} = R_k + jX_k \quad (3.17)$$

$$Y_{Tk} = j\omega C_k \quad (3.18)$$

Sendo R_k , X_k e C_k a resistência, a reatância e a capacitância totais da linha respetivamente.

Definindo:

$$G_k = \operatorname{Re} \left\{ \frac{1}{Z_{Sk}} \right\} = \frac{R_k}{R_k^2 + X_k^2} \quad (3.19)$$

$$B_k = \operatorname{Im} \left\{ \frac{1}{Z_{Sk}} \right\} = -\frac{X_k}{R_k^2 + X_k^2} \quad (3.20)$$

$$B'_k = \operatorname{Im} \left\{ \frac{Y_{Tk}}{2} \right\} = \frac{\omega C_k}{2} \quad (3.21)$$

Obtém-se as potências ativas e reativas, decompostas na parte real e imaginária:

junto ao nó i ,

$$P_{ij}^k = G_k V_i^2 - V_i V_j [G_k \cos(\theta_i - \theta_j) + B_k \sin(\theta_i - \theta_j)] \quad (3.22)$$

$$Q_{ij}^k = -(B_k + B'_k) V_i^2 - V_i V_j [G_k \sin(\theta_i - \theta_j) - B_k \cos(\theta_i - \theta_j)] \quad (3.23)$$

e junto ao nó j ,

$$P_{ji}^k = G_k V_j^2 - V_i V_j [G_k \cos(\theta_j - \theta_i) + B_k \sin(\theta_j - \theta_i)] \quad (3.24)$$

$$Q_{ji}^k = -(B_k + B'_k) V_j^2 - V_i V_j [G_k \sin(\theta_j - \theta_i) - B_k \cos(\theta_j - \theta_i)] \quad (3.25)$$

As perdas de potência ativa (P_L) e reativa (Q_L) obtêm-se somando as potências complexas que transitam na linha k do nó i para o j (S_{ij}^k) e do nó j para o i (S_{ji}^k),

$$S_{ij}^k + S_{ji}^k = P_{ji}^k + P_{ij}^k + j(Q_{ij}^k + Q_{ji}^k) = P_L + Q_L \quad (3.26)$$

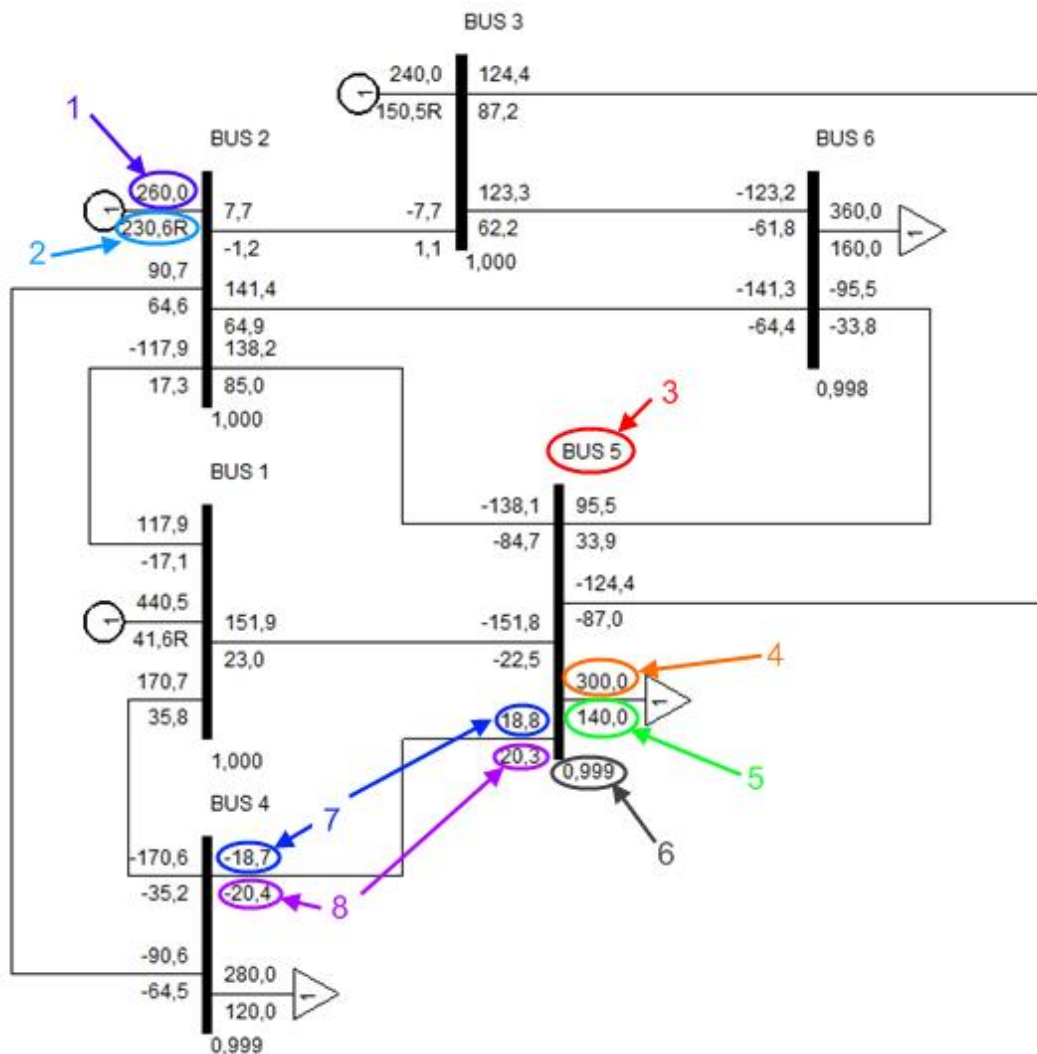
Substituindo as equações (3.22) a (3.25) na equação (3.26), resulta o valor das perdas P_L e Q_L para a linha k dada por:

$$P_L = G_k[V_i^2 + V_j^2 - 2V_iV_j\cos(\theta_i - \theta_j)] \quad (3.27)$$

$$Q_L = -(B_k + B'_k)(V_i^2 + V_j^2) + 2B_kV_iV_j\cos(\theta_i - \theta_j) \quad (3.28)$$

3.2.3 Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea” à rede de seis barramentos

Foi utilizada uma rede de 6 barramentos simplificada, baseada na rede de 6 barramentos, referida no IEEE [18], para o estudo deste método, cujos dados se encontram no apêndice A, ver Figura 12.



Legenda:

- 1 – Potência ativa gerada
- 2 – Potência reativa gerada
- 3 – Nome do barramento
- 4 – Potência ativa da carga
- 5 – Potência reativa da carga
- 6 – Tensão no barramento em pu
- 7 - Potência ativa que sai ou entra no barramento
- 8 - Potência reativa que sai ou entra no barramento

Figura 12 - Rede de 6-barramentos

Na Figura 13, é ilustrado o estado da rede em regime normal de funcionamento com indicação da percentagem de ocupação das linhas.

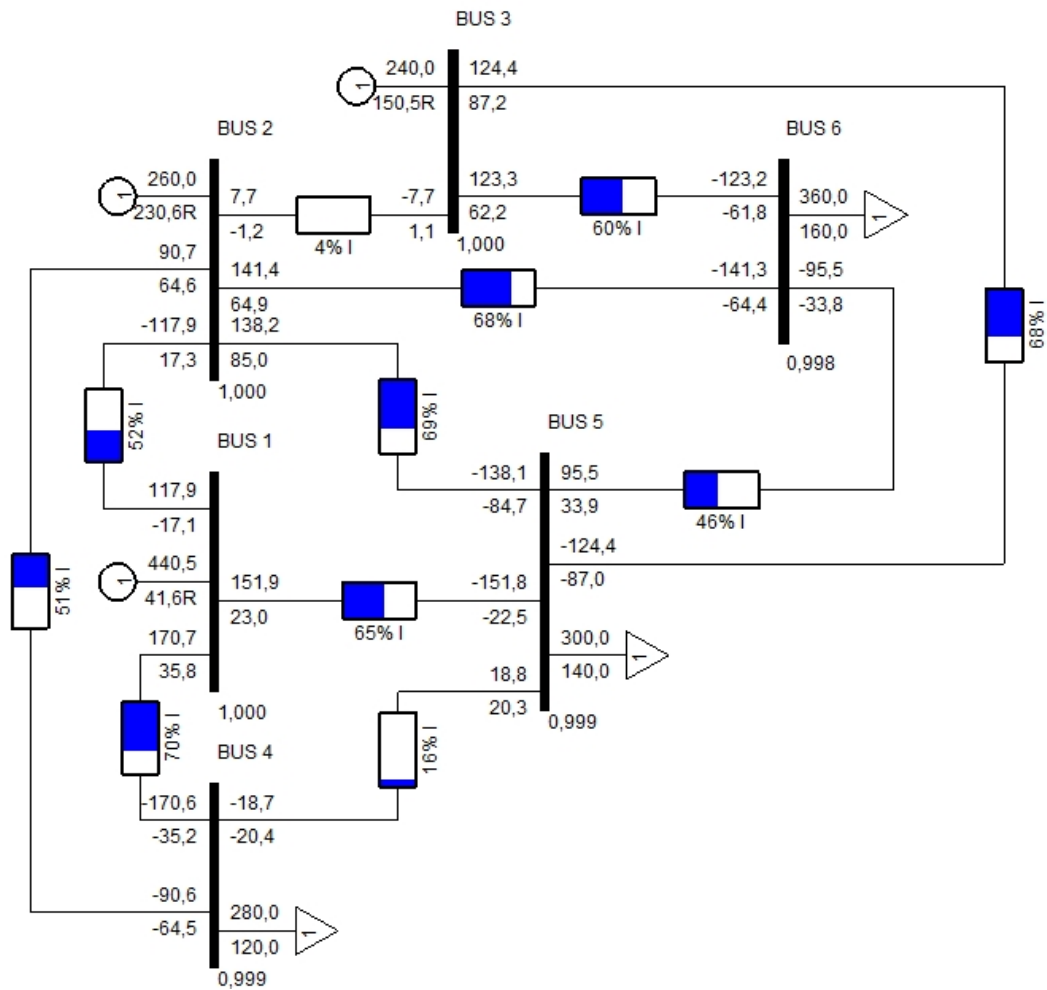


Figura 13 - Rede de 6-barramentos com indicação de ocupação de linha

Após executar o algoritmo “Máxima Injeção Nodal Não simultânea”, apresenta-se o caso onde é possível injetar a máxima potência num único nó, e cujo resultado se apresenta em baixo:

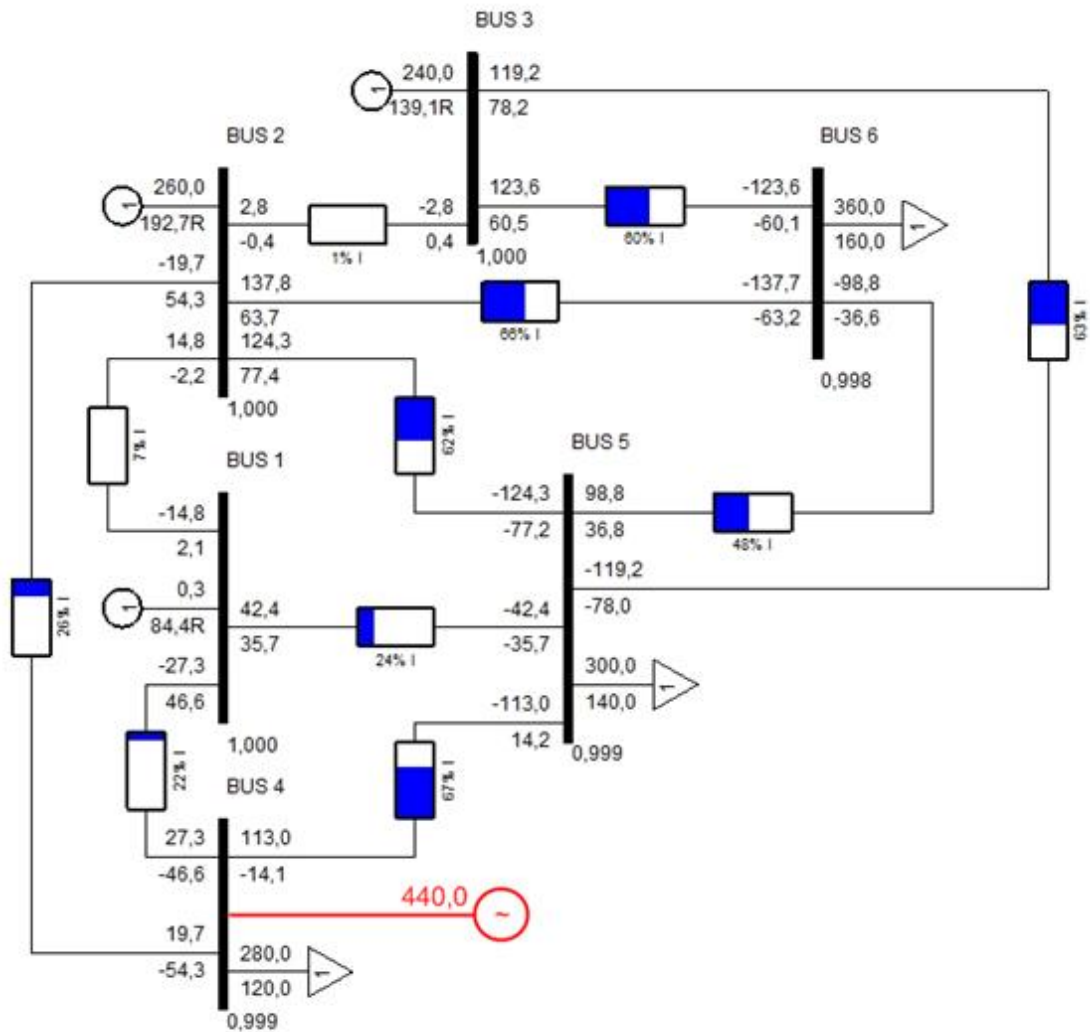


Figura 14 - Resultados após aplicação do algoritmo "Máxima Injeção Nodal Não simultânea" no nó 4

Na Figura 14, encontra-se ilustrado o caso no qual foi injetada potência no barramento 4, que originou um aumento de ocupação nas linhas ligadas a este barramento e um consequente reequilíbrio da rede. Neste caso, nenhuma das linhas se encontra com uma taxa de ocupação elevada porque o elemento limitador neste caso foi o gerador *Swing*, em que o seu valor chegou a zero.

No caso seguinte, foi injetada energia no barramento 2, que foi limitada pela linha 2-5, como é possível observar no seguinte esquema.

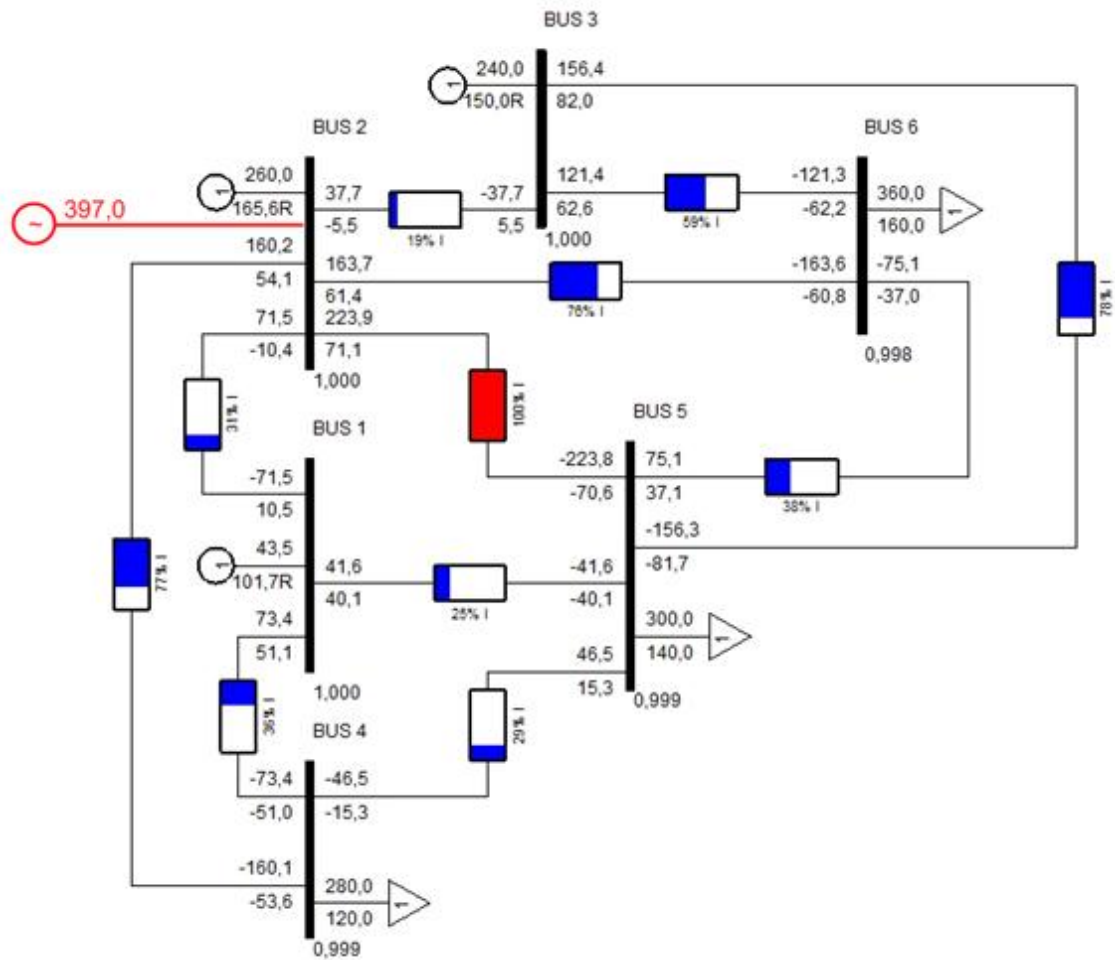


Figura 15 - Resultados após aplicação do algoritmo "Máxima Injeção Nodal Não simultânea" no nó 2

Após a execução deste algoritmo para todos os nós desta rede, chegou-se aos seguintes resultados:

Tabela 1 - Resultados do método "Máxima Injeção Nodal Não simultânea" aplicado à rede de 6 barramentos

Gerador no nó:	Potência Máxima de Injeção [MW]	Tipo de restrição
2	397	Linha do nó 2 para o 5 sobrecarregada
3	172	Linha do nó 3 para o 5 sobrecarregada
4	440	Potência gerada pelo gerador Swing chegou ao limite mínimo
5	440	Potência gerada pelo gerador Swing chegou ao limite mínimo
6	440	Potência gerada pelo gerador Swing chegou ao limite mínimo

Ao observar os resultados, pode-se concluir que os melhores locais para instalar novos grupos geradores seria nos nós 4, 5 ou 6. Se se pretendesse instalar geradores nos barramentos 2 ou 3, as linhas 2-5 e 3-5 teriam de ser melhoradas respetivamente, por forma a suportarem a nova geração.

Conclui-se que este método é um método simples e de certo modo “cego”, pois aumenta a potência injetada num determinado nó até ser atingido um dos limites impostos.

É preciso ter em consideração que o objetivo é saber a máxima capacidade de energia que é possível injetar na rede. Daí um dos limites ser não poder ultrapassar 100% da capacidade de qualquer uma das linhas. No entanto, este valor pode ser configurável, para por exemplo 70%, de modo a criar uma rede mais flexível.

3.2.4 Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea em Regime 'n-1'” à rede de seis barramentos

À semelhança do realizado em 3.2.3, usou-se a rede de 6 barramentos, apresentada na Figura 12.

Os resultados obtidos foram os seguintes:

Tabela 2 - Resultados do método "Máxima Injeção Nodal Não simultânea em Regime 'n-1'" quando aplicado à rede de 6 barramentos

Nó de injeção	Potência máxima que é possível injetar em regime N [MW]	Potência máxima que é possível injetar em regime N-1 [MW]	Elemento Limitador
2	397	30	Linha 2 - 5
3	172	8	Linha 3 - 5
4	440	440	Pg Gerador <i>Swing</i> = 0
5	440	351	Linha 4 - 5
6	440	181	Linha 3 - 5

Analisando os resultados, podemos concluir que no regime “N-1”, a potência máxima que é possível injetar na rede é menor.

A rede em regime de contingência N-1 (uma das linhas está fora de serviço) tem de suportar a mesma energia, mas com menos um elemento de transporte (linha). Logo, a energia presente nas restantes linhas é maior, o que implica que a energia adicional que é possível injetar na rede pode, em geral, ser menor.

Quando é injetada energia no nó 4 não se verifica esta diminuição de potência injetada porque a injeção de energia é limitada pelo gerador *Swing* e não pelas linhas.

3.2.5 Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável” à rede de seis barramentos

Após aplicar o algoritmo, como descrito em 3.1.2, à rede de 6 barramentos, em que os geradores 1, 2 e 3 funcionam como gerador *Swing* sucessivamente, obteve-se os seguintes resultados:

Tabela 3 - Resultados do método "Máxima Injeção Nodal Não simultânea com Localização de Swing Variável" aplicado à rede de 6 barramentos em regime N

Swing	Gerador 1		Gerador 2		Gerador 3	
	Nós	PG máx	Elemento limitador	PG máx	Elemento limitador	PG máx
1	-		229	Linha 1 - 2	240	Linha 1 - 5
2	397	Linha 2 - 5			240	Pg Gerador <i>Swing</i> = 0
3	172	Linha 3 - 5	204	Linha 3 - 5		
4	440	Pg Gerador <i>Swing</i> = 0	259	Pg Gerador <i>Swing</i> = 0	239	Pg Gerador <i>Swing</i> = 0
5	440	Pg Gerador <i>Swing</i> = 0	259	Pg Gerador <i>Swing</i> = 0	239	Pg Gerador <i>Swing</i> = 0
6	440	Pg Gerador <i>Swing</i> = 0	259	Pg Gerador <i>Swing</i> = 0	239	Pg Gerador <i>Swing</i> = 0

Através dos resultados apresentados na Tabela 3, verifica-se que, ao alternar o funcionamento como gerador *Swing* do gerador 1 para o 2 e 3, as potências máximas de injeção em cada nó são diferentes.

Observando o nó 4, a potência máxima que é possível injetar neste nó é 440, 259 e 239 MW para o cenário com o gerador 1, 2 e 3 como gerador *Swing*, respetivamente. Assim, a potência máxima que é possível injetar num determinado nó depende do gerador que funciona como gerador *Swing*.

No nó 1 verifica-se uma alteração no funcionamento da rede, em que o elemento limitador deste nó é a linha 1-2, quando o gerador 2 é o gerador *Swing*, e a linha 1-5, quando o gerador 3 é o gerador *Swing*. Assim, o gerador *Swing* tem impacto ao nível do funcionamento da rede e, como tal, é essencial na projeção futura da rede, em que é necessário identificar quais os elementos que necessitam de melhorias para a rede poder acomodar mais energia no futuro.

Concluindo, a localização do gerador como gerador *Swing* na rede tem impacto ao nível do funcionamento da mesma, levando a níveis diferentes de capacidade máxima de potência que é possível injetar na rede.

3.2.6 Aplicação do algoritmo “Máxima Injeção Nodal Não simultânea em Regime ‘n-1’ e com Localização de Swing Variável” à rede de seis barramentos

O algoritmo, apresentado em 3.1.3, quando aplicado à rede de 6 barramentos, gerou os seguintes resultados.

Tabela 4 - Resultados do método "Máxima Injeção Nodal Não simultânea em Regime ‘n-1’ e com Localização de Swing Variável" aplicado à rede de 6 barramentos

Swing	Gerador 1		Gerador 2		Gerador 3	
	Nós	PG máx	Elemento limitador	PG máx	Elemento limitador	PG máx
1	-		14	Linha 1 - 5	12	Linha 1 - 5
2	30	Linha 2 - 5			46	Linha 2 - 5
3	8	Linha 3 - 5	9	Linha 3 - 5		
4	440	Pg Gerador <i>Swing</i> = 0	113	Linha 1 - 2	136	Linha 1 - 5
5	351	Linha 4 - 5	143	Linha 1 - 2	209	Linha 5 - 6
6	181	Linha 3 - 5	250	Linha 1 - 2	239	Pg Gerador <i>Swing</i> = 0

Através dos resultados apresentados na Tabela 5, verifica-se que, à semelhança do algoritmo “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável”, ao alternar o funcionamento como gerador *Swing*, do gerador 1 para o 2 e 3, as potências máximas de injeção em cada nó são diferentes. Assim, a potência máxima que é possível injetar num determinado nó depende do gerador que funciona como gerador *Swing*, quer se esteja ou não a analisar a rede em regime de contingência. Os valores apresentados para cada localização do Gerador *Swing* foram todos inferiores em regime N-1, exceto no caso em que a limitação se deveu ao gerador *Swing*.

3.3 ANÁLISE E COMPARAÇÃO ENTRE OS VÁRIOS ALGORITMOS

Ao longo das seções anteriores, foram apresentados métodos para o cálculo da potência máxima que é possível injetar na rede a partir de um único nó, injeção não simultânea.

A tabela seguinte apresenta os valores quando aplicados estes métodos à rede de 6 barramentos.

Tabela 5 - Potência máxima de injeção individual na rede por nó tendo em os vários métodos

Potência Máxima que é possível injetar num nó de acordo com os métodos indicados				
Injeção no nó:	"Máxima Injeção Nodal Não simultânea" (Gerador 1 é o Swing)	"Máxima Injeção Nodal Não simultânea em Regime 'n-1'"	"Máxima Injeção Nodal Não simultânea com Localização de Swing Variável" (Gerador 2 é o Swing)	"Máxima Injeção Nodal Não simultânea com Localização de Swing Variável" (Gerador 3 é o Swing)
1			229	240
2	397	30		240
3	172	8	204	
4	440	440	259	239
5	440	351	259	239
6	440	181	259	239

Tal como foi referido nas seções anteriores, comparando o método "Máxima Injeção Nodal Não simultânea" e "Máxima Injeção Nodal Não simultânea em Regime 'n-1'", podemos verificar que a rede em regime N-1 (em que uma das linhas está desativada) permite um menor valor de potência injetada. O que é de esperar, uma vez que, ao colocar uma linha fora de serviço, está-se a limitar a transmissão da energia. Logo, vai haver menor capacidade de transporte e possibilidades de caminhos para a mesma energia gerada, levando a um aumento de sobrecarga das linhas. Assim, a energia adicional que é possível injetar na rede também é menor.

Esta análise verifica-se para todos os casos em que o funcionamento como gerador Swing vai alternando entre os vários geradores existentes na rede, como é possível verificar na tabela a seguir.

Tabela 6 – Comparação entre os métodos de Swing variável em regime N e N-1 aplicados à rede de 6 barramentos

Nós	Gerador 1 como <i>Swing</i>		Gerador 2 como <i>Swing</i>		Gerador 3 como <i>Swing</i>	
	Em regime N	Em regime N-1	Em regime N	Em regime N-1	Em regime N	Em regime N-1
1			229	14	240	12
2	397	30			240	46
3	172	8	204	9		
4	440	440	259	113	239	136
5	440	351	259	143	239	209
6	440	181	259	250	239	239

Relativamente ao método “Máxima Injeção Nodal Não simultânea com Localização de Swing Variável”, verifica-se que o modo de funcionamento da rede altera-se quando outro gerador se comporta como gerador *Swing*, o que vai alterar a capacidade máxima de potência que é possível injetar na rede, como se pode verificar na Tabela 3 e Tabela 6. Quando é injetada energia no nó 4, a potência máxima que é possível injetar neste nó é diferente, pressupondo que o gerador 1, 2 ou 3 é o gerador *Swing*. O mesmo se verifica para o método “Máxima Injeção Nodal Não simultânea em Regime ‘n-1’ e com Localização de Swing Variável”, ver Tabela 4 e Tabela 6.

Um aspeto a ter em atenção com o método “Máxima Injeção Nodal Não simultânea” é o facto deste método, apesar de apenas indicar o máximo de potência que é possível injetar num único nó, permitir indicar o elemento limitador nesta situação. Assim, caso se pretenda instalar uma nova geração num nó específico, podemos verificar qual o elemento limitador que é necessário alterar, de modo a permitir a alocação da nova energia na rede.

CAPÍTULO IV – INJEÇÕES SIMULTÂNEAS

Apresentação de algoritmos para o cálculo máximo de injeção nodal simultânea.

4. INJEÇÕES SIMULTÂNEAS

4.1 MÉTODO HEURÍSTICO

No capítulo anterior foi demonstrado um método capaz de calcular a potência máxima que é possível injetar num determinado nó. No entanto, este método só indica a capacidade da rede para uma injeção não simultânea num único nó.

O método apresentado neste capítulo pretende apresentar um método para o cálculo da potência máxima que é possível injetar na rede, ou seja, em vários nós simultaneamente, pois a potência máxima que é possível injetar num nó pode nem sempre ser igual à potência máxima que é possível injetar na rede.

O método heurístico, tal como indica a sua definição, é um método de aproximação que tenta encontrar soluções robustas para o problema, tendo em consideração a simultaneidade das injeções, a sua localização e montante injetado.

Para um determinado cenário de carga e topologia da rede, começa-se por injetar energia num nó até aparecer uma linha em sobrecarga. Esta, passa a ser o limite máximo que a linha consegue suportar quando for injetada energia neste nó. De seguida, é relaxada a restrição que representa a capacidade da linha, de modo a não entrar em sobrecarga, e continua-se a injetar energia no nó até outra linha exceder o limite de capacidade e assim sucessivamente. O objetivo é identificar as linhas que limitam a injeção neste nó e em que valores isso sucede.

Aplica-se este método a todos os nós da rede e determina-se a potência máxima associada ao nó para as linhas limitadoras que entram em sobrecarga.

Após a obtenção dos limites de potência máxima para cada linha, tendo em conta os nós onde foi injetada a energia, calcula-se a potência simultânea que é possível injetar em todos os nós da rede.

Neste capítulo são apresentadas e testadas duas variantes do método heurístico, tendo em consideração diferentes fatores de simultaneidade associados às injeções nodais:

1 – O fator de simultaneidade da primeira variante é calculado da seguinte forma:

$$\text{Factor de simultaneidade}_{\text{ramo } x} = \left(\frac{\lambda_1}{\lambda_1}\right) + \left(\frac{\lambda_1}{\lambda_2}\right) + \left(\frac{\lambda_1}{\lambda_3}\right) + \dots \quad (4.1)$$

2 – O fator de simultaneidade da segunda variante é calculado da seguinte forma:

$$\text{Factor de simultaneidade}_{\text{ramo } x} = \left(\frac{\lambda_1}{\lambda_1}\right)^{1,0} + \left(\frac{\lambda_1}{\lambda_2}\right)^{1,2} + \left(\frac{\lambda_1}{\lambda_3}\right)^{1,4} + \dots \quad (4.2)$$

O fator de simultaneidade é calculado por linhas e tem em conta as potências injetadas que levam as linhas a entrar em sobrecarga.

λ_1 corresponde à potência mais baixa que é possível injetar até o ramo 1 entrar em sobrecarga, independentemente do nó onde é injetada a energia. λ_2 é o segundo valor mais baixo, correspondente à potência injetada antes de o ramo 2 entrar em sobrecarga. Assim, sucessivamente para todas as potências associadas ao ramo 1.

Após o cálculo do fator de simultaneidade para todas as linhas, divide-se o valor máximo que é possível injetar em cada nó pelos vários fatores de simultaneidade associados a cada ramo e escolhe-se o que tem o valor mais baixo por nó. O valor máximo que é possível injetar em cada nó é calculado pelo método “Máxima Injeção Nodal Não simultânea”, já o valor mais baixo, solução final, é o valor final correspondente à máxima injeção simultânea que é possível injetar nesse nó na rede.

O fator de simultaneidade é calculado de modo a que o valor final incida mais sobre a potência associada ao primeiro elemento limitador. Tendo, assim, o primeiro elemento limitador mais relevância/peso. Deste modo, dá-se maior ênfase às restrições que impõem uma potência injetada menor, ou seja, eletricamente mais próximas.

Relativamente à segunda variante do fator exponencial, esta permite reduzir o “peso/impacto” das restrições mais afastadas eletricamente.

Exemplo:

O exemplo seguinte ilustra o cálculo do método heurístico.

Considere-se uma rede elétrica genérica com 4 barramentos. A tabela seguinte ilustra a potência injetada num determinado nó e respetiva potência quando uma linha entra em sobrecarga.

Tabela 7 - Resultados intermédios do exemplo do método heurístico

Nó de injeção	Capacidade de injeção relativa à linha em sobrecarga [MW]			
	Linha 1-2	Linha 1-4	Linha 2-3	...
1	380	500	430	...
2	400	820	370	...
3	600	1100	800	...
4	800	410	900	...

O próximo passo consiste no cálculo do fator de potência para cada elemento limitador através das equações (4.1) (4.2).

Ex: cálculo do fator de simultaneidade para a linha 1-2

1ª variante do fator de simultaneidade

$$Factor\ de\ simultaneidade_{ramo\ x} = \left(\frac{380}{380}\right) + \left(\frac{380}{400}\right) + \left(\frac{380}{600}\right) + \left(\frac{380}{800}\right) = 3,058$$

2ª variante do fator de simultaneidade

$$Factor\ de\ simultaneidade_{ramo\ x} = \left(\frac{380}{380}\right)^{1,0} + \left(\frac{380}{400}\right)^{1,2} + \left(\frac{380}{600}\right)^{1,4} + \left(\frac{380}{800}\right)^{1,6} = 2,772$$

Após o cálculo do fator de simultaneidade para cada elemento limitador, aplica-se o fator de simultaneidade às potências determinadas no passo anterior, encontradas na Tabela 7. Encontram-se os seguintes resultados,

Tabela 8 - Resultados intermédios do exemplo do método heurístico - após aplicação do fator de simultaneidade

Variante do fator de simultaneidade	Capacidade de injeção relativa à linha em sobrecarga [MW] após aplicação dos fatores de simultaneidade						
	1ª	2ª	1ª	2ª	1ª	2ª	
Nó de injeção	Linha 1-2		Linha 1-4		Linha 2-3		...
1	124	137	186	199	157	176	...
2	131	144	305	327	135	152	...
3	196	216	409	439	293	328	...
4	262	289	152	164	329	369	...
Fator de simultaneidade aplicado	3.058	2.772	2.693	2.506	2.734	2.441	...

A tabela seguinte compara os valores da injeção máxima de potência para o nó 1, tendo em conta a injeção individual e a injeção simultânea baseada no método heurístico com a 1ª variante do fator de simultaneidade.

Tabela 9 - Resultados finais do exemplo do método heurístico

Nó de injeção	Capacidade de injeção relativa à linha em sobrecarga [MW] após aplicação dos fatores de simultaneidade						
	Linha 1-2		Linha 1-4		Linha 2-3		...
1	380	124	500	186	430	157	...
...
Variante do fator de simultaneidade	1ª	2ª	1ª	2ª	1ª	2ª	

Analisando os resultados obtidos, verifica-se através da Tabela 9 que o valor máximo que é possível injetar no nó 1 não simultaneamente é 380 MW e o valor máximo que é possível injetar no nó 1 simultaneamente é 124MW. Este valor corresponde ao valor mais baixo resultante da execução do método heurístico, quando aplicado a primeira variante do fator de simultaneidade.

O diagrama de blocos seguinte ilustra os passos envolvidos para a aplicação deste método.

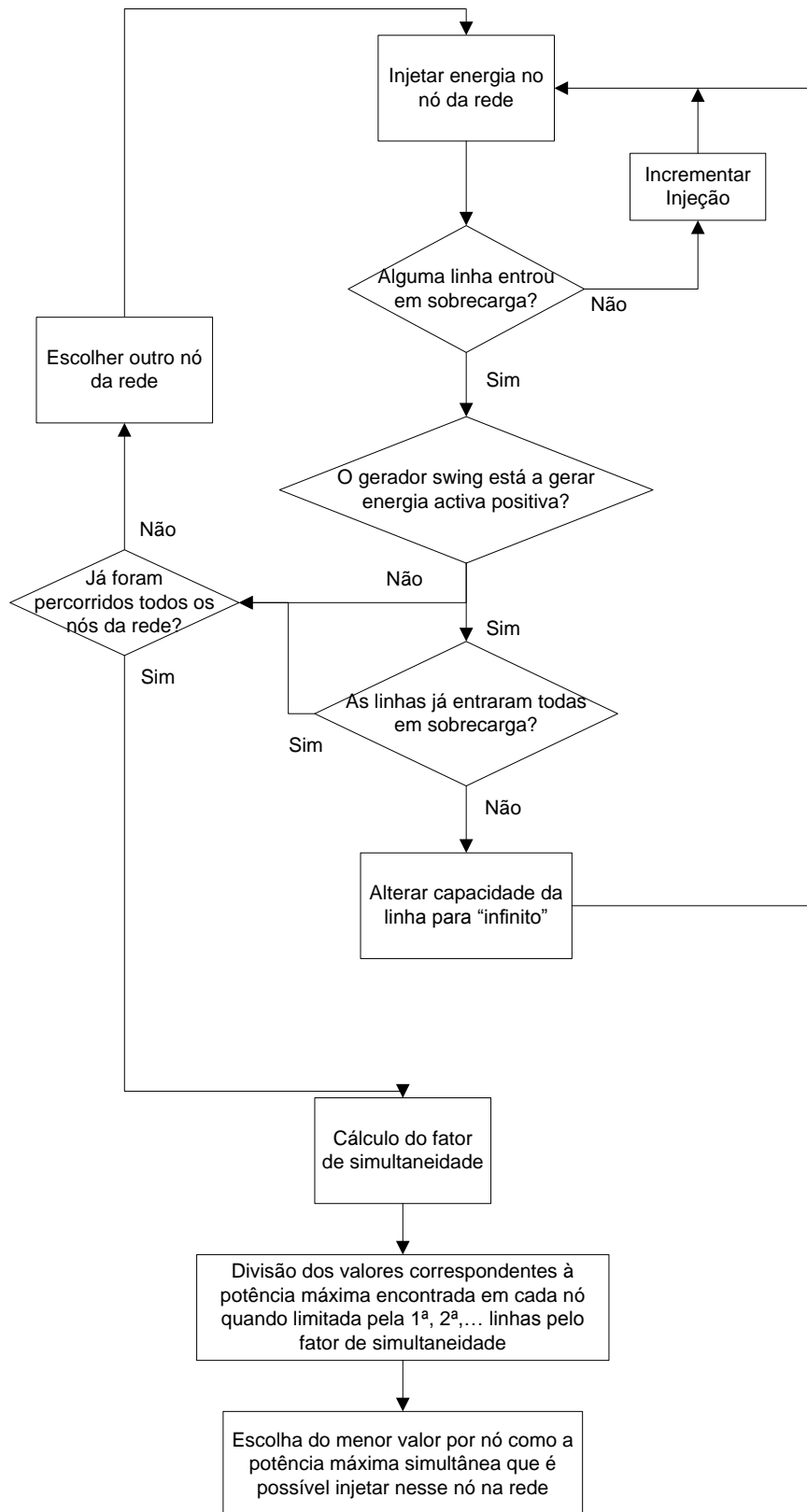


Diagrama 5 - Descrição do método Heurístico

4.2 ALGORITMO GENÉTICO

Face aos algoritmos apresentados, existe a necessidade de desenvolver um algoritmo capaz de explorar outras combinações de geração e que forneça soluções exequíveis, de modo a maximizar a injeção de energia simultânea que é possível injetar em todos os nós da rede.

Dada a natureza combinatória e discreta do problema, procedeu-se ao desenvolvimento de um algoritmo genético, adaptado ao problema.

O algoritmo genético é um método de pesquisa e de otimização que tenta, de certo modo, imitar o processo natural de evolução. Este método, muito utilizado na ciência de Computação para achar soluções aproximadas de problemas, foi inicialmente fundamentado pelo americano John Henry Holland. O algoritmo genético, como foi referido anteriormente, pertence ao grupo de algoritmos evolutivos que, tal como o nome indica, utilizam técnicas inspiradas na biologia evolutiva, como a hereditariedade, mutação, seleção natural e recombinação ou cruzamento.

O algoritmo genético procura encontrar uma solução perto do ótimo, ao longo de vários ciclos, os quais são designados de gerações. O algoritmo começa por criar, aleatoriamente, um conjunto de soluções válidas que vão constituir a primeira geração. Este conjunto de indivíduos é submetido ao processo evolutivo, constituído pelas seguintes etapas, para formar a nova geração, a qual é utilizada como ponto de partida na próxima iteração do algoritmo.

Codificação - Um indivíduo é constituído por genes, em que cada gene corresponde à injeção de energia num nó da rede. Assim, analisando a Figura 16, relativamente ao indivíduo A, o primeiro gene indica uma injeção de 20MW no primeiro barramento, o segundo gene indica uma injeção de 33 MW e, assim, sucessivamente.

20	33	12	77	18	23	29	44	Indivíduo A
16	13	15	14	8	3	9	7	Indivíduo B

Figura 16 – Exemplo de codificação de 2 indivíduos

Etapas do algoritmo genético:

Avaliação – Os indivíduos (soluções) da geração são avaliados. Neste caso particular é feita uma análise com o PSSE para verificar se as soluções apresentadas correspondem ao requerido e se cumprem a limitações impostas. As limitações impostas consistem em não haver linhas com ocupação superior a 100% do seu rate e a potência gerada pelo gerador *Swing* não ser negativa;

Exemplo: Após verificação dos indivíduos A e B pelo PSSE ambos são válidos.

Seleção – Os indivíduos são selecionados aleatoriamente para a reprodução;

Exemplo: Os indivíduos A e B foram selecionados para a reprodução.

Reprodução:

Cruzamento – As características dos indivíduos selecionados são recombinadas, gerando um novo indivíduo, isto é, os indivíduos são escolhidos dois a dois. De seguida, é escolhido um ponto pivô aleatoriamente que separa os genes dos indivíduos. A partir destes dois indivíduos, são criados dois descendentes com uma parte dos genes de um indivíduo e com a outra parte do outro indivíduo, partes essas definidas pelo ponto pivô;

Exemplo: Dos indivíduos A e B, foi escolhido um pivô aleatoriamente e foram gerados 2 descendentes, que se apresentam como os indivíduos C e D.

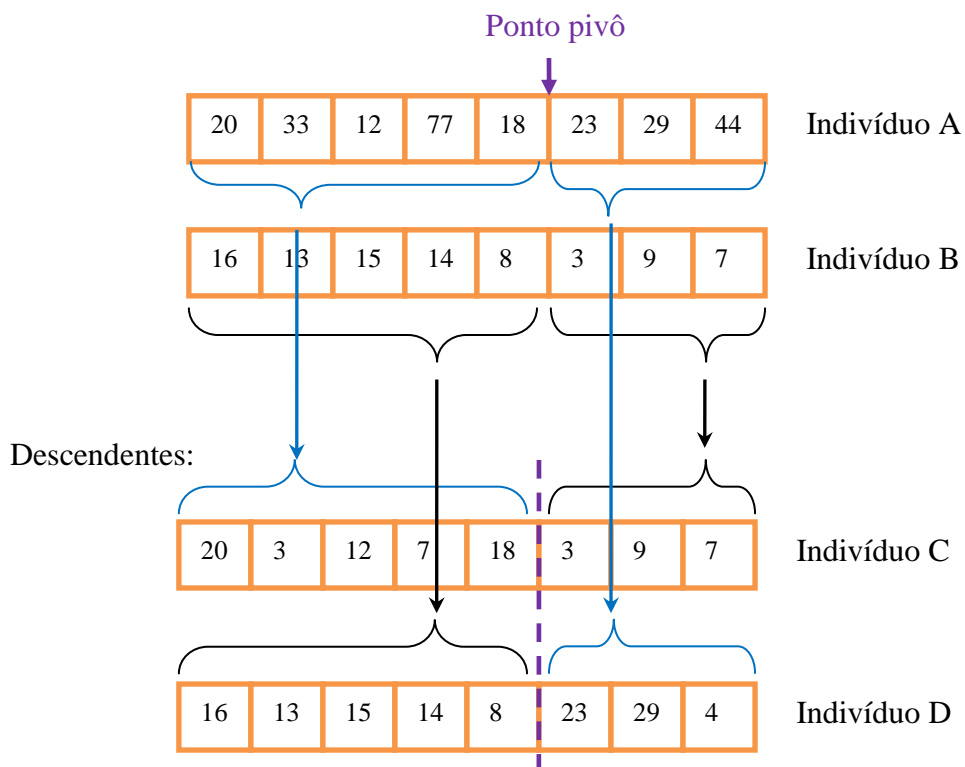


Figura 17 – Exemplo da etapa de cruzamento do algoritmo genético

Mutação – A mutação indica uma etapa do processo evolutivo onde as características dos indivíduos são alteradas, acrescentando assim variedade à população. O processo de mutação tem uma probabilidade muito baixa de ocorrer e só ocorre num gene de cada vez.

- Proposta de operador de mutação:

No algoritmo genético implementado neste trabalho é definida uma mutação adaptada ao problema. Ou seja, quando um indivíduo obtém a probabilidade de sofrer mutação, executa-se o método “Máxima Injeção Nodal Não simultânea”, tendo em conta as características do

indivíduo em questão. Assim, sabendo a potência que é possível injetar no nó correspondente a esse gene e a sua potência atual, o valor da mutação, apesar de aleatório, vai estar compreendido entre o valor atual e o valor máximo que é possível colocar para que a solução seja válida. Deste modo, temos um fator de mutação que tende sempre para o ótimo;

Exemplo: Assumindo que o indivíduo D vai sofrer mutação no gene número 5.

Após a aplicação do método “Máxima Injeção Nodal Não simultânea” no barramento 5 da rede correspondente ao indivíduo D (ver Figura 17), assumindo que o valor máximo que o gene número 5 pode ter é 34, o resultado da mutação para o gene 5 é um valor aleatório entre o número atual e número máximo que este gene pode tomar, tendo em conta a rede correspondente ao indivíduo, ou seja, um valor entre 8 e 34. Exemplo 26. Assim, o indivíduo D resultaria em,

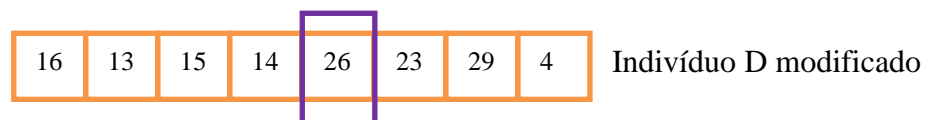


Figura 18 – Resultado final da etapa de mutação aplicada ao indivíduo D

Elitismo – Os 2 melhores indivíduos são preservados de modo a garantir que a melhor solução é passada de geração em geração.

Exemplo: Considerando apenas os indivíduos A, B, C e D, os elitistas seriam os indivíduos A e D, pois a soma dos valores correspondentes às potências injetadas (valor da função fitness – função objetivo) são maiores nos indivíduos A e D, como é possível verificar a seguir.

									Valor da função Fitness	
									↓	
Indivíduo A	20	3	12	7	18	23	29	4	=	116
Indivíduo B	16	13	15	14	8	3	9	7	=	85
Indivíduo C	20	3	12	7	18	3	9	7	=	79
Indivíduo D	16	13	15	14	8	23	29	4	=	122

Figura 19 – Escolha dos melhores indivíduos na etapa Elitismo

Os indivíduos resultantes do processo de reprodução são inseridos na população e, desta nova população, são escolhidos aleatoriamente os indivíduos que vão constituir a população final, de modo a que o tamanho da população seja sempre o mesmo.

Finalização – Nesta etapa é verificado se as condições de finalização do algoritmo genético foram atingidas, retornando para a etapa de avaliação em caso negativo ou finalizando em caso positivo. As condições de finalização do algoritmo genético consistem num número máximo de gerações e na saturação da população.

O algoritmo genético otimiza uma população de indivíduos de acordo com a sua função objetivo. Assim, ao tentar criar o melhor indivíduo, uma vez que este corresponde à injeção de potência simultânea numa rede, está-se a tentar achar a melhor disposição de injeção de energia na rede que se pretende otimizar.

Exemplo: Assumindo que o indivíduo D (ver Figura 17) é o melhor indivíduo criado pelo algoritmo genético, a rede correspondente traduz-se na seguinte injeção simultânea adicional na rede:

Tabela 10 - Exemplo de uma rede a partir dos dados de um indivíduo do algoritmo genético

Barramento	Injeção adicional na rede [MW]
1	16
2	13
3	15
4	14
5	8
6	23
7	29
8	4

No diagrama seguinte encontra-se ilustrado o funcionamento deste método.

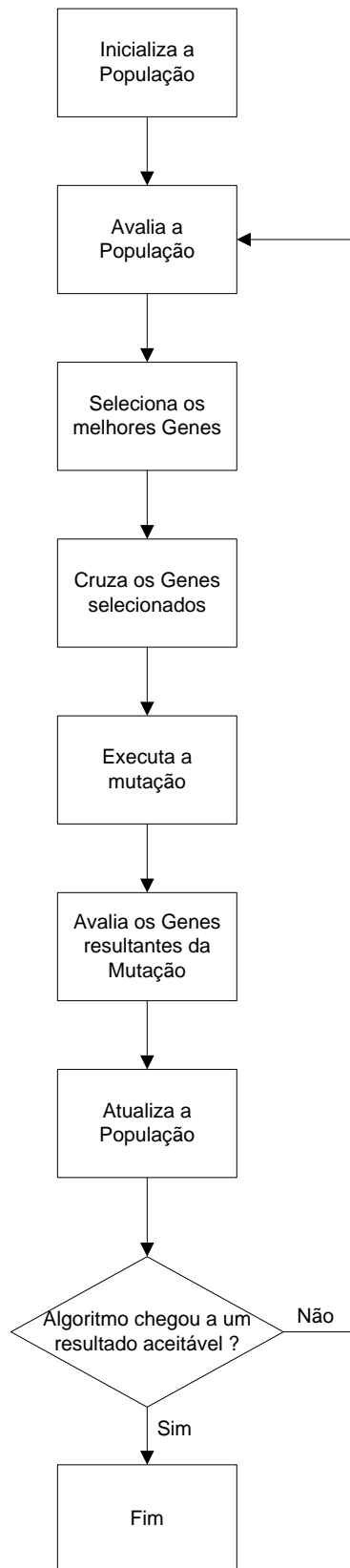


Diagrama 6 - Descrição do algoritmo Genético

4.3 IMPLEMENTAÇÃO DOS ALGORITMOS

A estrutura do código desenvolvido sofreu uma grande alteração, de modo a poder acomodar os novos algoritmos, ver Figura 20.

Ou seja, para além das novas funções que tiveram de ser criadas para a execução dos algoritmos “Método Heurístico” e “Algoritmo Genético”, as estruturas de suporte/alocação de dados foram alteradas. As estruturas tornaram-se, assim, mais complexas, pois no caso dos algoritmos não simultâneos apenas se usa uma rede de cada vez. No entanto, no algoritmo genético o número de redes e dados respetivos é igual ao número da população, sendo estes dados dinâmicos e variáveis de rede para rede. No caso de uma população de 40 indivíduos, a estrutura tem de suportar os dados e a sua manipulação de 40 redes ao mesmo tempo.

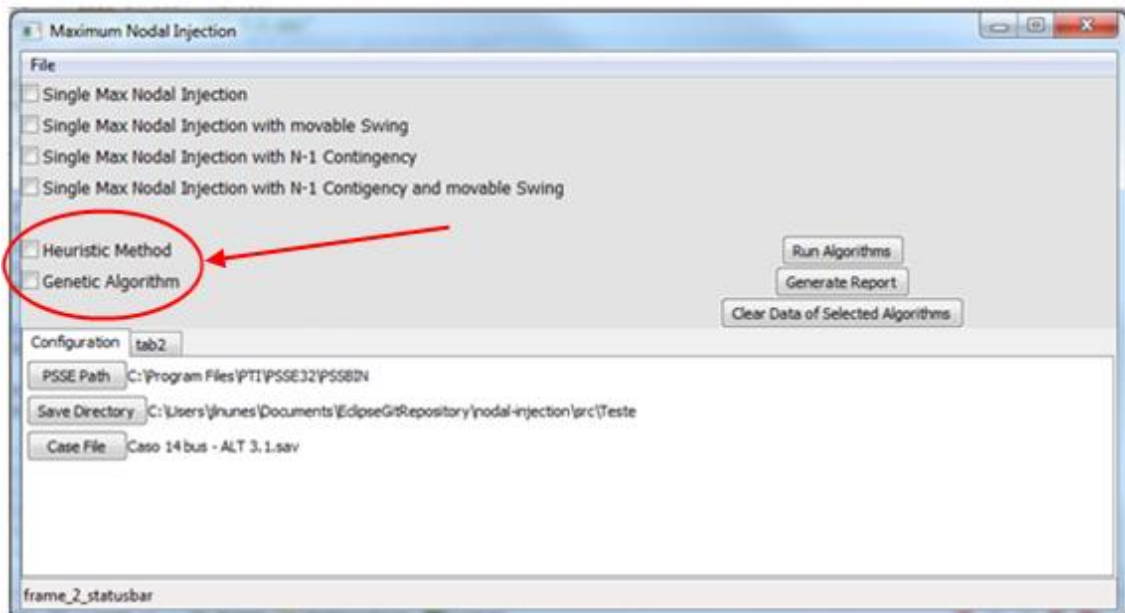


Figura 20 - Interface gráfico da aplicação desenvolvida

4.3.1 Aplicação do algoritmo Heurístico à rede de seis e catorze barramentos

O método Heurístico foi implementado na rede de 6 barramentos, apresentada nos capítulos anteriores. Na aplicação desta heurística, os resultados não foram os esperados, pois apenas foi possível calcular, para esta rede, o fator de simultaneidade para o nó 2 e 3. Não foi possível calcular o fator de simultaneidade para os restantes nós, uma vez que a potência gerada pelo gerador Swing chega a zero antes de qualquer uma das linhas chegar a uma taxa de ocupação de 100%. Logo, não é possível calcular o fator de simultaneidade de acordo com o método descrito. Assim, consideraram-se os valores obtidos para os nós 2 e 3 (141MW e 96 MW, respetivamente) e para os restantes nós (4, 5 e 6) e dividiu-se a restante energia que é gerada pelo gerador Swing (203MW) pelos 3 restantes nós, tendo em conta que este valor não pode ser maior que um terço da máxima potência injetada por estes nós.

A máxima potência injetada nos nós 4, 5 e 6 é 440MW (ver Tabela 1), potência limitada pelo gerador Swing, da qual um terço é 146,67 MW. Como a restante potência gerada pelo gerador swing é 203,4 MW, quando é injetada a potência calculada nos nós 2 e 3, um terço de 203,4MW é 67,8MW, logo, $67,8\text{MW} < 146,67\text{MW}$. Assim, foi injetada a potência adicional de 67,8MW nos nós 4,5 e 6, ver Figura 22.

De seguida, apresenta-se a rede após a aplicação do método heurístico, com e sem a adição da injeção nos nós 4, 5, e 6.

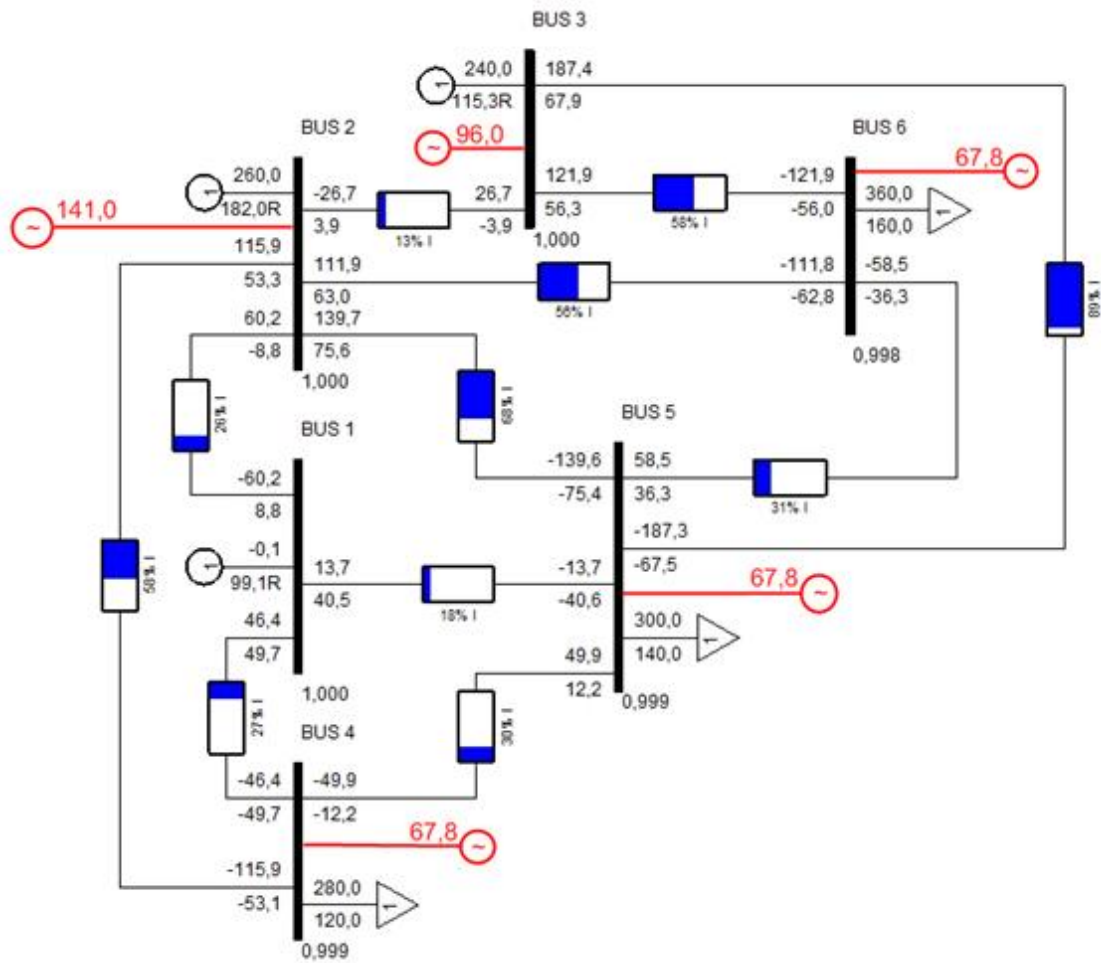


Figura 22 – Resultados obtidos após aplicação do "Método Heurístico" com alteração

A tabela seguinte contém os resultados do método heurístico para a rede de 6 barramentos, com e sem a alteração final descrita anteriormente que consiste em injetar energia nos nós 4, 5 e 6.

Tabela 11 - Resultados obtidos quando aplicado o método Heurístico à rede de 6 barramentos, tendo e não em conta as alterações definidas

Nó	Potência injetada adicional [MW] (1)		Potência injetada adicional [MW] (2)	
	Sem alteração	Com alteração	Sem alteração	Com alteração
2	142	142	144	144
3	97	97	108	108
4	0	67,8		62,7
5	0	67,8	0	62,7
6	0	67,8	0	62,7

- (1) – Cálculo da injeção de potência através do método heurístico com recurso à primeira variante do fator de simultaneidade.
- (2) – Cálculo da injeção de potência através do método heurístico com recurso à segunda variante do fator de simultaneidade.

Ao aplicar o método heurístico a esta rede, conclui-se que o mesmo não consegue calcular todos os valores de injeção simultânea para todos os nós devido à restrição do gerador *Swing*. O que significa que, para esta rede e nas condições de carga e geração indicadas, a rede não apresenta limitações nos seus ramos em virtude de injeções nodais simultâneas.

Assim, testa-se o método heurístico para uma rede com outras características, neste caso para uma rede de 14 barramentos. Averigua-se se, pelo facto de se ter uma rede diferente, as linhas entram em sobrecarga antes da potência gerada pelo gerador *Swing* chegar a zero, de modo a se poder calcular o fator de simultaneidade.

A rede de 14 barramentos usada foi baseada na rede IEEE 14-BUS [19] e apresenta-se nas figuras seguintes, com e sem indicação da taxa de ocupação das linhas.

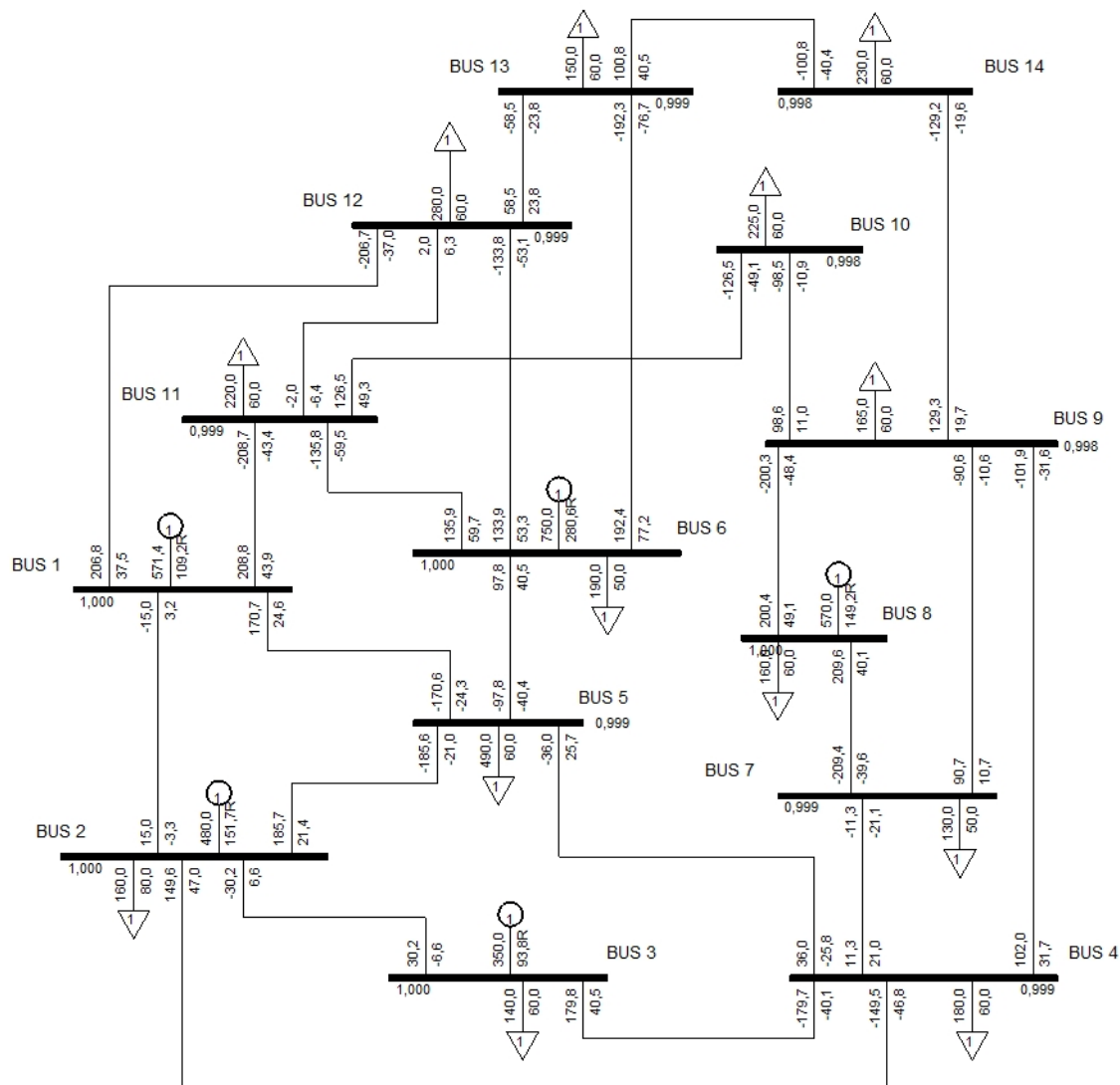


Figura 23 – Rede de 14 barramentos sem indicação de taxa de ocupação de linhas

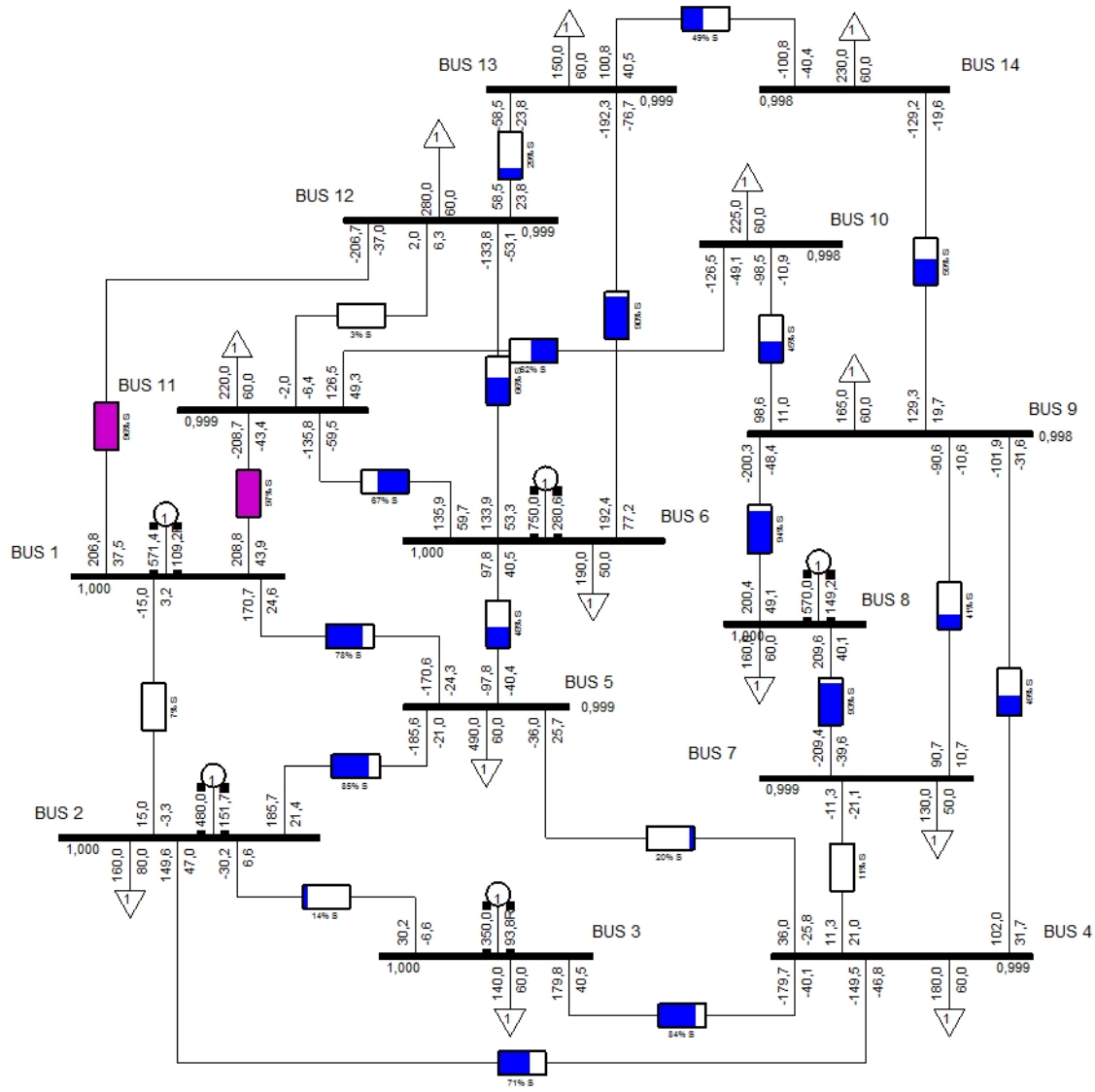


Figura 24 - Rede de 14 barramentos com indicação de taxa de ocupação de linhas

Após aplicar o método heurístico a esta rede, os resultados obtidos foram os seguintes:

Tabela 12 - Resultados obtidos quando aplicado o método Heurístico à rede de 14 barramentos, tendo em conta as duas variantes de cálculo do factor de simultaneidade

BUS	Potência injetada adicional [MW] (1)	Potência injetada adicional [MW] (2)
1	-	-
2	72	79
3	61	68
4	187	189
5	-	-
6	105	118
7	77	83
8	22	24
9	154	160
10	-	-
11	-	-
12	-	-
13	188	189
14	-	-
15	-	-

- (1) – Cálculo da injeção de potência através do método heurístico, com recurso à primeira variante do fator de simultaneidade.
- (2) – Cálculo da injeção de potência através do método heurístico, com recurso à segunda variante do fator de simultaneidade.

Com os seguintes valores de Potência gerada pelo gerador Swing:

	Potência Ativa gerada (tendo em conta a variante 1 do fator de simultaneidade)	Potência Ativa gerada (tendo em conta a variante 2 do fator de simultaneidade)
Gerador Swing	-296 MW	-338 MW

Cujas redes se traduzem em,

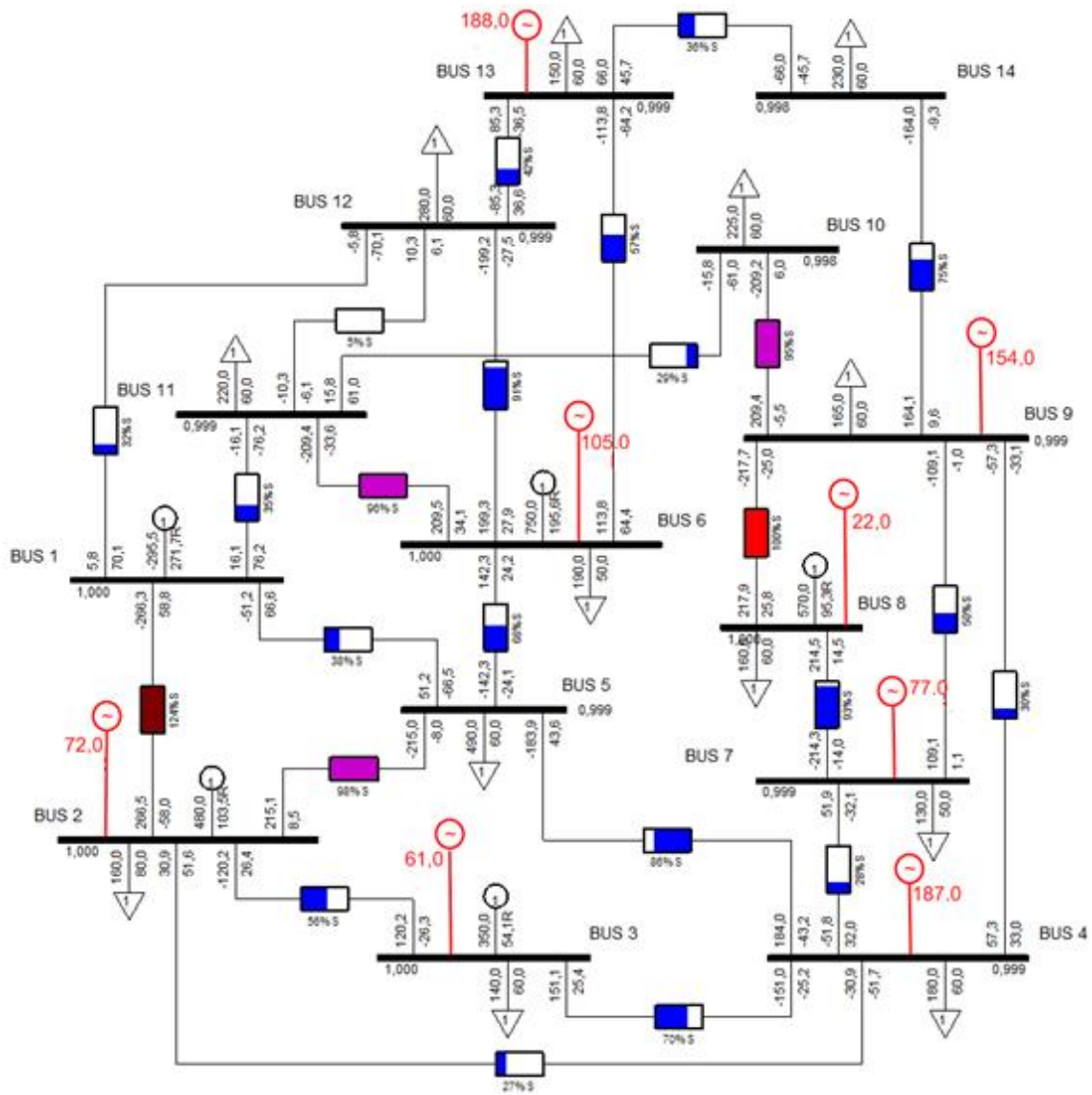


Figura 25 - Rede com aplicação do método Heurístico utilizando a variante do fator de simultaneidade 1

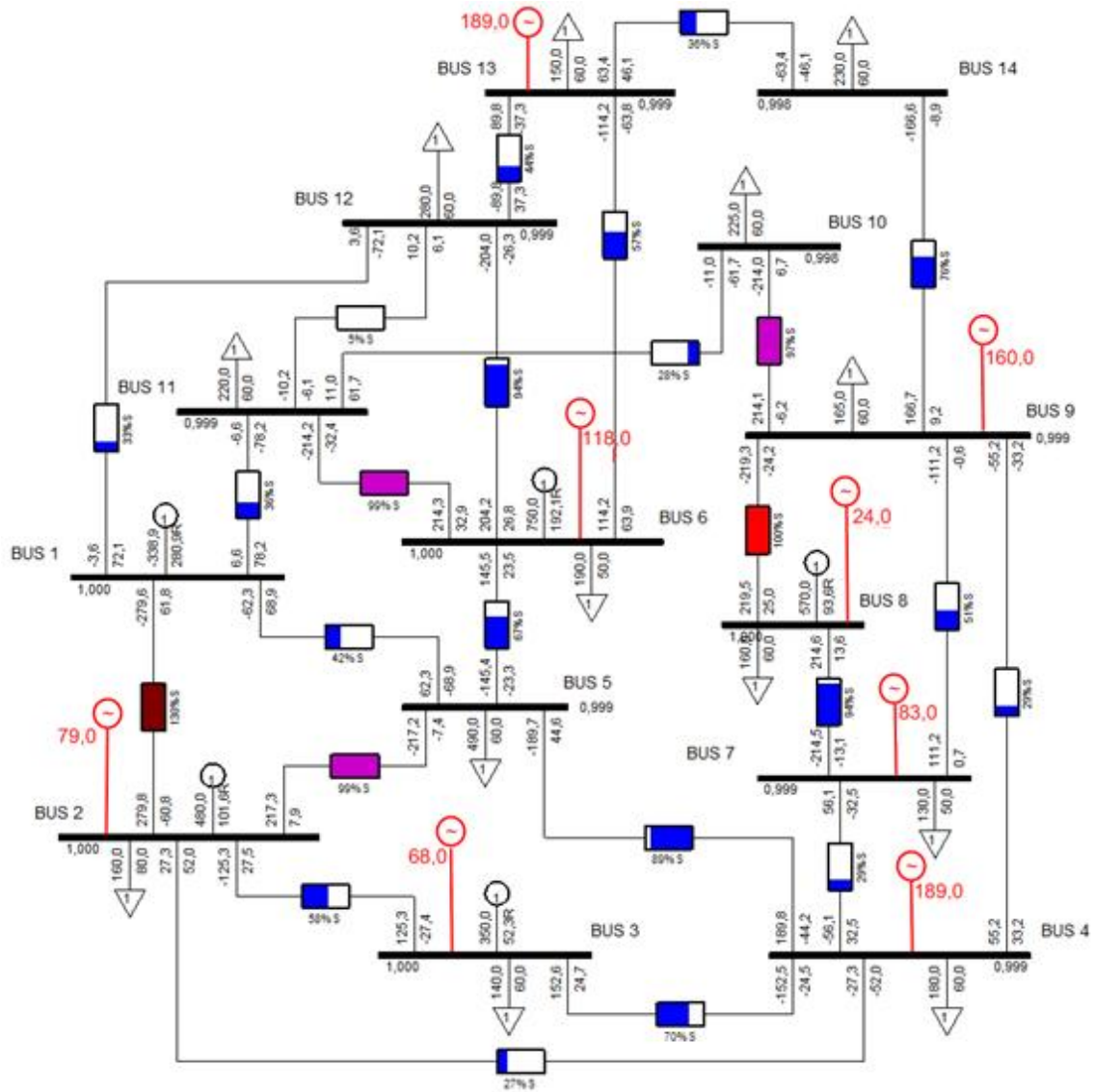


Figura 26 - Rede com aplicação do método Heurístico utilizando a variante do fator de simultaneidade 2

Observando as redes, verifica-se que existe uma evolução dos resultados quando comparados com os da rede de 6 nós. Ou seja, é possível obter mais resultados para as potências de injeção simultâneas. Observa-se, igualmente, que o método, apesar de indicar valores para a injeção simultânea, apresenta valores que não levam a uma rede plausível, nem a um resultado desejável. Nos casos apresentados em cima, as linhas estão em sobrecarga e a potência gerada pelo gerador Swing é negativa.

No entanto, a rede apresenta valores que permitem observar as limitações da rede e que serve de um ponto de partida para o estudo de uma geração

distribuída. Até porque, em situações reais, nunca se aplica estas injeções todas em simultâneo.

A topologia da rede representa uma fator determinante. Redes mais carregadas ou mais radiais em torno do nó que se injeta são redes em que esta metodologia melhor capta a simultaneidade das injeções.

Fatores como o *Swing* variável, outros cenários e condições de operação (ex.: regime N-1), como explorado e demonstrado no capítulo III, não foram objeto do presente estudo.

Não obstante, o funcionamento e resultados chegados levam a supor que este método seja mais adequado no cálculo da geração simultânea de um número específico de nós, parte integrante de uma rede maior, onde a geração neste conjunto específico de nós pouco influencia a potência gerada pelo gerador *Swing*.

4.3.2 Aplicação do algoritmo genético à rede de 6 e 14 barramentos

Tendo em conta a rede de 6 barramentos, foi aplicado o algoritmo genético, cujos resultados se apresentam na seguinte tabela:

Tabela 13 – Melhor indivíduo aplicado à rede de 6 barramentos tendo em conta a geração presente na rede

BUS	Algoritmo Genético
1	-
2	20 (+260)
3	20 (+240)
4	18
5	61
6	4 (+ 150)
Total	440

Nota: O valor entre parêntesis corresponde à geração já presente na rede.

A figura seguinte traduz estes resultados.

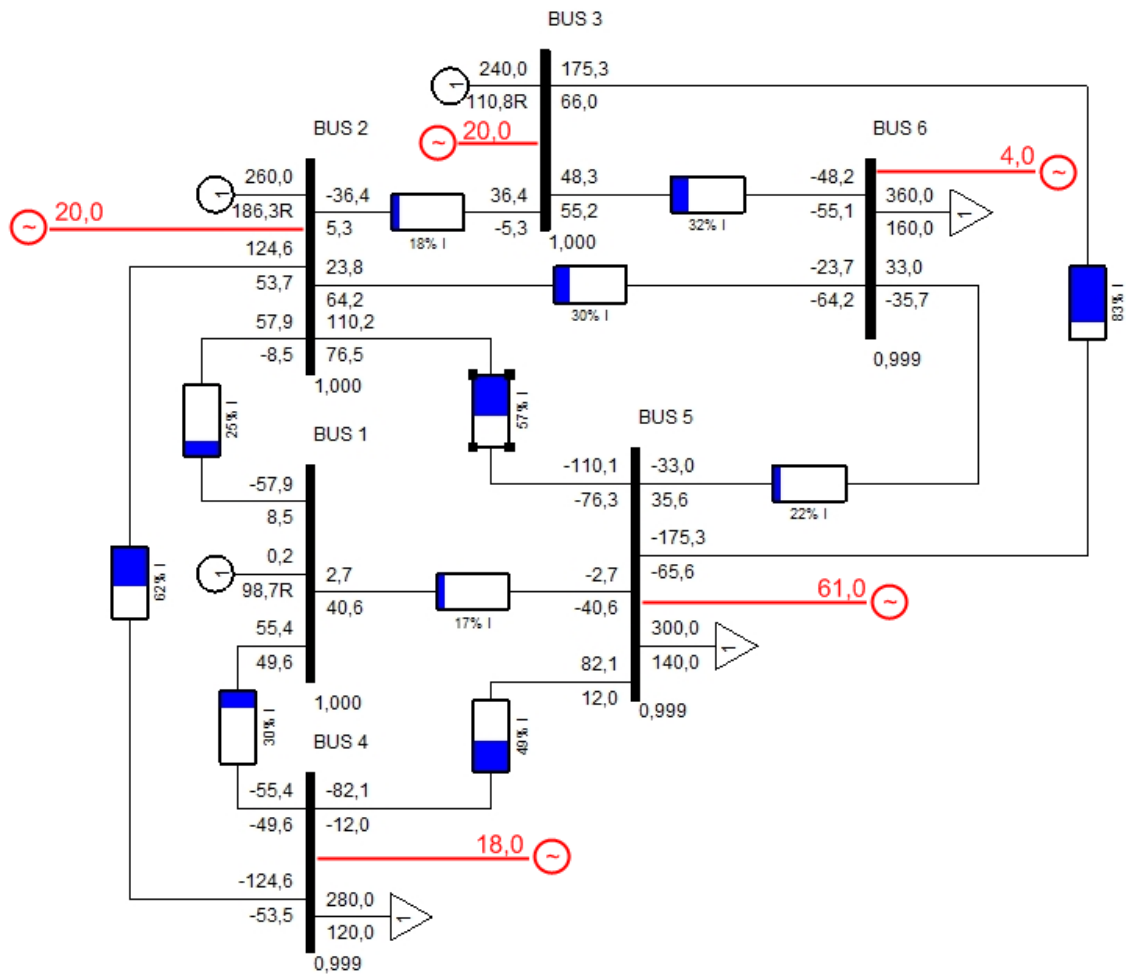


Figura 27 - Algoritmo genético aplicado à rede de 6 barramentos tendo em conta a geração existente

Após ser aplicado o algoritmo genético, verifica-se que esta rede não leva a situações que não cumpram as restrições impostas, as quais são as sobrecargas das linhas e a potência gerada pelo gerador Swing positiva. No caso acima apresentado, o algoritmo genético chegou a um indivíduo em que a injeção simultânea é limitada pela geração do Swing, neste caso nula, limite inferior imposto.

Os resultados apresentados indicam que se pode injetar mais energia no nó 6 do que nos outros. Esta situação não homogénea deve-se ao facto de já existir geração pré-existente.

No caso seguinte, o algoritmo genético é aplicado à rede de 14 barramentos, tendo em conta a geração existente.

Os resultados obtidos pelo algoritmo genético, quando aplicado à rede de 14 barramentos, apresentam-se a seguir, considerando-se a geração já existente.

Tabela 14 - Melhor indivíduo aplicado à rede de 14 barramentos tendo em conta a geração presente na rede

BUS	Algoritmo Genético – melhor indivíduo tendo em conta a geração existente (valor entre parêntesis)
1	-
2	8 (+480)
3	13 (+350)
4	15
5	263
6	8 (+750)
7	58
8	11 (+570)
9	19
10	15
11	101
12	16
13	26
14	18
Total	571 (+2150) = 2721

Cuja rede se traduz em:

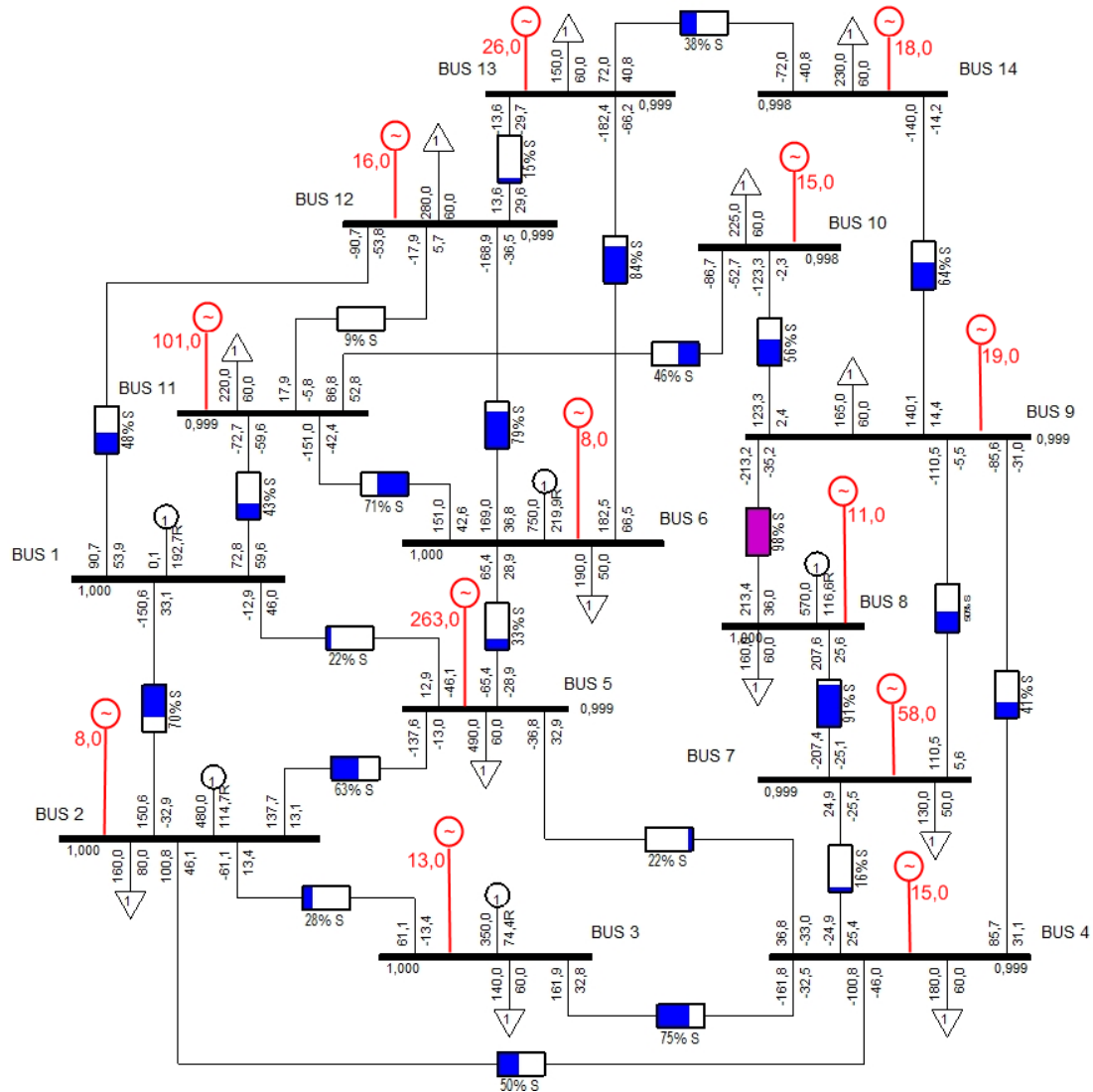


Figura 28 - Algoritmo genético aplicado à rede de 14 barramentos tendo em conta a geração existente

Com o objetivo de verificar a capacidade do algoritmo genético em distribuir a potência injetada na rede, foi aplicado este método à rede de 14 barramentos, mas não considerando a geração existente.

A tabela seguinte contém os resultados obtidos.

Tabela 15 - Melhor indivíduo aplicado à rede de 14 barramentos não tendo em consideração a geração presente na rede

BUS	Algoritmo Genético – melhor indivíduo não tendo em consideração a geração existente
1	-
2	245
3	231
4	231
5	106
6	127
7	215
8	17
9	236
10	237
11	244
12	149
13	433
14	249
Total	2720

Cuja rede se traduz em:

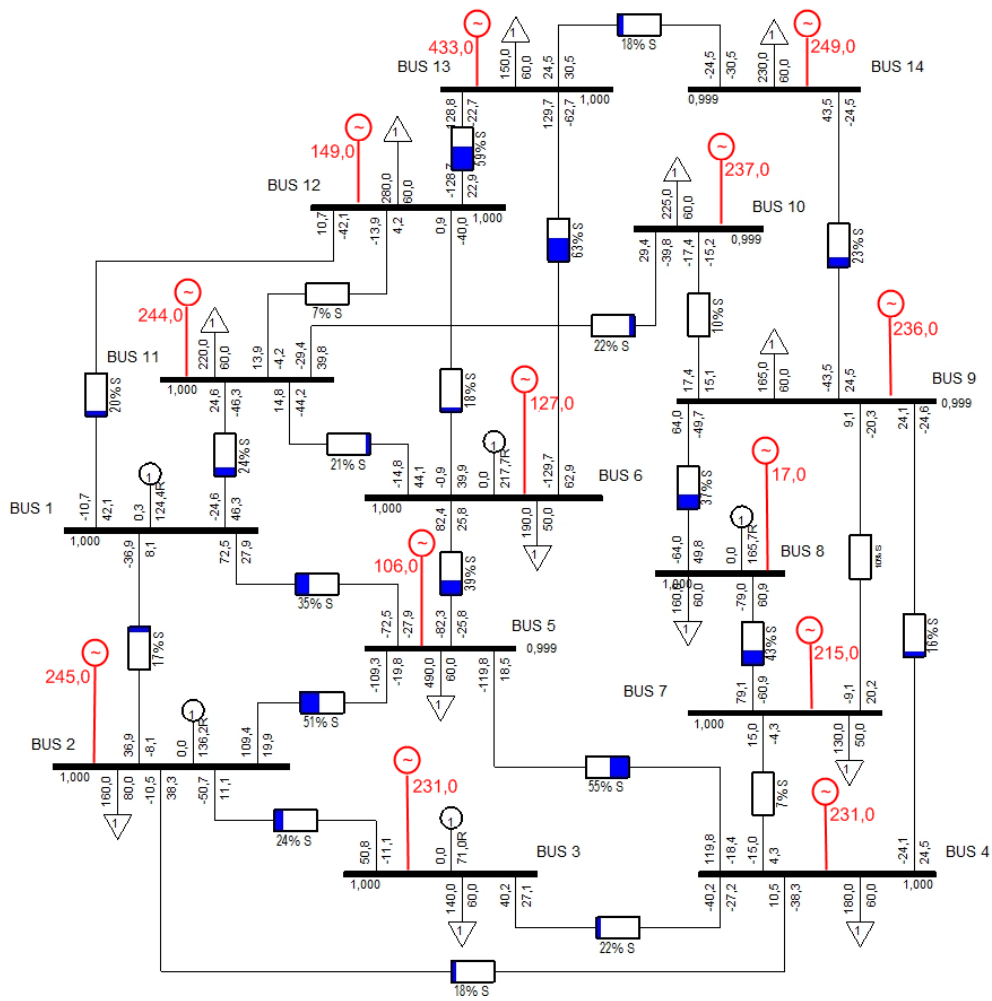


Figura 29 – Algoritmo genético aplicado à rede de 14 barramentos não tendo em consideração a geração existente

Consegue-se, claramente, concluir que, se não se tiver em consideração a geração existente, o algoritmo genético tende a distribuir a energia gerada perto do consumo, levando a uma diminuição substancial da sobrecarga das linhas. O que já não acontece no primeiro caso, em que, uma vez que temos em conta a geração inicialmente produzida, o algoritmo genético tenta encontrar o melhor indivíduo que consiga distribuir a energia de uma forma mais eficiente, tendo sempre em conta a energia produzida pelos geradores já presentes na rede e as restrições impostas.

Em ambos os casos, o elemento limitador é o gerador Swing que atingiu uma geração de 0MW.

Os parâmetros descritos na Tabela 16 foram os escolhidos, tendo em conta os resultados de vários testes.

A percentagem de 85% na etapa de cruzamento garante uma boa taxa de variação de novos indivíduos, fruto dos indivíduos que foram cruzados.

Quanto à mutação, esta permite obter valores fora dos existentes na população, levando a uma solução única e diferente. O facto de se aplicar uma mutação inteligente e adaptativa permite orientar o resultado da mutação para um indivíduo melhor.

Os parâmetros utilizados na configuração do algoritmo genético são:

Tabela 16 - Parâmetros de configuração dos algoritmos genéticos aplicados

Parâmetro	Valor
Numero Máximo de gerações	40
Tamanho da População	30
Número de indivíduos elitistas por geração	2
Percentagem de Cruzamento entre indivíduos	85%
Percentagem de Mutação	8,5%

Ao longo das gerações do algoritmo genético, a média da população tende a aumentar. A evolução da média da população dos dois casos acima apresentados para a rede de 14 barramentos é apresentada nos gráficos seguintes.

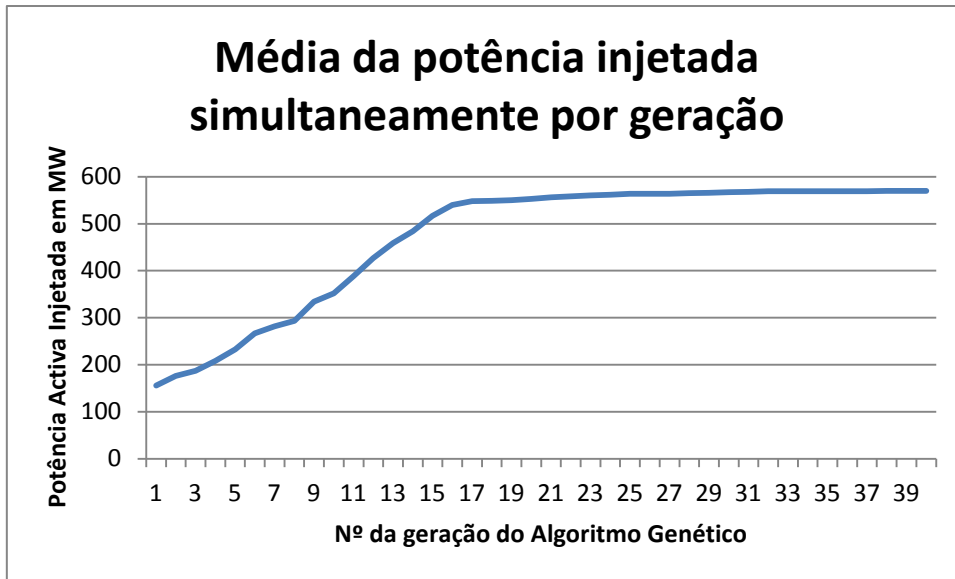


Gráfico 1 - Média de potência injetada na rede simultaneamente tendo em conta a geração existente

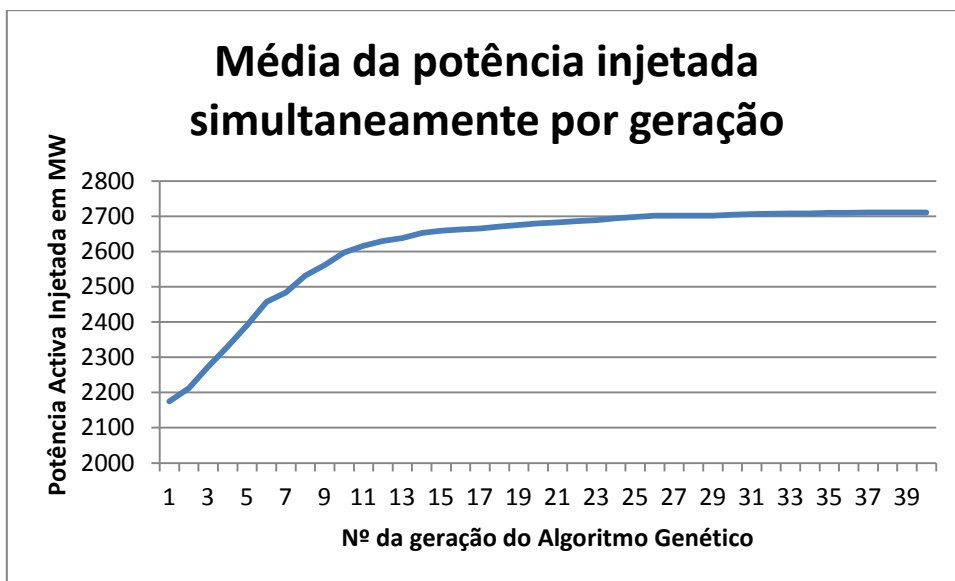


Gráfico 2 - Média de potência injetada na rede simultaneamente não tendo em conta a geração existente

Observando os gráficos, Gráfico 1 e Gráfico 2, pode-se verificar que, em ambas as situações, o método do algoritmo genético evolui em média ao longo das gerações.

Note-se que, no Gráfico 1, os valores são bastante menores porque só se tem em conta as potências injetadas adicionais, ou seja, não contempla os valores da geração já existente.

Ao introduzir a ideia de elitismo, consegue-se conservar os dois melhores indivíduos de uma geração para a geração seguinte, garantindo que o melhor indivíduo produzido durante o algoritmo Genético não é perdido ao longo das gerações.

O algoritmo Genético é um método que consegue determinar a potência máxima que é possível injetar numa rede, indicando os valores da potência a injetar em cada nó. Através da solução obtida, torna-se possível identificar os elementos mais apropriados à instalação de nova geração, bem como os *upgrades* necessários a fazer aos elementos que restrinjam a instalação de uma nova geração num determinado nó.

4.4 ANÁLISE E COMPARAÇÃO ENTRE OS VÁRIOS ALGORITMOS

Os resultados obtidos a partir dos métodos Heurístico e Genético, quando aplicados a rede de 14 Bus, encontram-se na seguinte tabela:

Tabela 17 - Potência máxima de injeção simultânea na rede

Número do nó	Método Heurístico	Algoritmo Genético
2	72	8
3	61	13
4	187	15
5	-	263
6	105	8
7	77	58
8	22	11
9	154	19
10	-	15
11	-	101
12	-	16
13	188	26
14		18
Máxima Energia adicional que é possível injetar	867 – 295 (valor negativo gerado pelo Swing) = 571	571

A partir da tabela anterior, é possível observar que ambos os métodos chegam ao mesmo resultado. No entanto, no método Heurístico, o resultado final não é válido, uma vez que algumas linhas entram em sobrecarga e que o gerador Swing comporta-se como uma carga ao apresentar um valor negativo de geração.

Pode-se, então, concluir que o algoritmo Genético é mais vantajoso porque permite obter um resultado válido e exequível. De facto, não tem qualquer limitação em relação ao tipo (mais ou menos radial) ou dimensão de rede onde é aplicado, permitindo, ainda, a possibilidade de configurar o limite máximo de ocupação das linhas. Neste trabalho foi aplicada uma percentagem de ocupação de 100% como limite máximo, uma vez que se pretende obter a potência máxima que é possível injetar na rede.

Comparando o método “Máxima Injeção Nodal Não simultânea” com o algoritmo Genético, o algoritmo Genético apresenta resultados melhores, uma vez que permite a injeção simultânea. No entanto, no caso de se injetar energia num nó, aplicando o método “Máxima Injeção Nodal Não simultânea”, o elemento limitador for a Potência gerada pelo gerador Swing, o resultado seria muito semelhante ao algoritmo genético. Com efeito, não é possível injetar mais energia na rede do que a produzida pelo gerador Swing, já que uma das restrições impostas é a potência gerada pelo gerador Swing não poder ser negativa. Os valores apenas seriam diferentes devido às perdas de energia nas linhas por transmissão.

Convém mencionar que as redes apresentadas foram analisadas para um determinado cenário de carga / geração. Ou seja, as redes apresentadas não foram testadas para outros cenários, como perfis de geração (Verão / Inverno) ou de carga (ponta / vazio), nem para condições de operação de rede, como seja considerar o regime N-1.

CAPÍTULO V - CONCLUSÃO

5. CONCLUSÃO

5.1 OBSERVAÇÕES FINAIS

Foram desenvolvidos, no presente trabalho, métodos e algoritmos para determinar a capacidade máxima de recepção de potência nos nós de uma rede.

O conhecimento da potência máxima possível de injetar numa determinada rede é crucial e de extrema importância para o planeamento e gestão de uma rede. Efetivamente, a partir destes dados é possível estimar e gerir a colocação de novos centros produtores, bem como projetar futuras alterações na rede.

Torna-se, então, evidente o interesse no desenvolvimento de algoritmos capazes de determinar soluções para este problema.

Tendo em conta o problema como base de partida, foram propostas, nos capítulos 3 e 4, várias metodologias para a sua solução, cujos resultados foram apresentados e comentados ao longo dos mesmos capítulos.

No capítulo 3 foram desenvolvidos métodos baseados em buscas gaussianas para a determinação da máxima injeção nodal numa rede não simultaneamente.

Verificou-se que o valor calculado é sensível em relação à localização do nó de balanço e ao regime de topologia de operação (regime N-1).

Para a resolução do problema das injeções simultâneas, foram apresentadas heurísticas que têm em consideração o carácter local da injeção. Tratando-se de um problema com uma natureza combinatória, foi desenvolvido um novo método baseado em algoritmos genéticos.

Conclui-se que a aplicação de Algoritmos Genéticos afigura-se como uma solução técnica robusta, a qual permite a obtenção de soluções técnicas de elevada qualidade sem, no entanto, pôr em causa o desempenho da rede.

Conclui-se que a obtenção de valores de capacidade nodal da potência máxima de uma rede é um problema combinatório complexo, atendendo à necessidade de garantir a simultaneidade, bem como ao facto de que qualquer solução de combinações de valores de injeção ter de ser viável do ponto de vista de exploração do sistema / rede. O algoritmo genético, como foi referido no capítulo 4, apresenta-se como o algoritmo que garante soluções fiáveis e exequíveis, soluções estas que foram determinadas após análise e teste de centenas, senão milhares, de possibilidades num curto espaço de tempo, tornando-se, por isso, numa ótima ferramenta para o estudo e projeto de redes elétricas.

5.2 PERSPECTIVAS DE TRABALHO FUTURO

No decurso deste projeto, foi desenvolvido um *software* de interface visual (capítulo 3.1), de modo a permitir o uso dos algoritmos desenvolvidos por pessoas sem conhecimento de programação. Este *software*, para além de fornecer os resultados referentes a cada algoritmo, cria igualmente as soluções finais, ou seja, cria redes que se encontram em formato compatível com o *software* PSS/E.

Propõe-se, assim, complementar o interface visual do utilizador com parâmetros que só estão acessíveis através do código de programação.

Sugestões de desenvolvimento para a continuação e melhoramento do trabalho:

- A implementação de outros métodos de otimização, como por exemplo o algoritmo “Particle Swarm Optimization”, para comparação com o algoritmo Genético apresentado nesta tese;
- A adaptação do algoritmo genético, de forma a estimar capacidades de injeção zonal;
- A inclusão de aspetos funcionais na construção das soluções (ex: regime N-1, critérios de planeamento, ...);
- A implementação da funcionalidade adicional no algoritmo genético, de modo a escolher os nós que se pretendem estudar, ao invés da rede inteira, permitindo o estudo de uma região específica da rede. Esta possibilidade é conseguida aplicando apenas os nós desejados na construção dos indivíduos do método genético;
- A aplicação do Algoritmo Genético a redes reais e para diferentes cenários de carga/geração.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] - Harrison, G. P., Wallace, A. R. (2005), "Optimal Power Flow Evaluation of Distribution Network Capacity for the Connection of Distributed Generation", Proceedings Institute Electric Engineers - Generation, Transmission and Distribution, Vol. 152, no. 1, Janeiro de 2005, páginas 115 – 122.
- [2] - Harrison, G. P., Wallace, A. R., "Planning for Optimal Accommodation of Dispersed Generation in Distribution Networks", The University of Edinburgh - United Kingdom.
- [3] - Harrison, G. P., Piccolo, A., Siano, P., Wallace, A. R. (2007), "Distributed Generation Capacity Evaluation Using Combined Genetic Algorithm and OPF", International Journal of Emerging Electric Power Systems, vol. 8, Issue 2, 2007.
- [4] – Karegar, H. K., Jalilzadeh, S., Nabaci, V., Shabani, A., "Reconfiguration of Deregulated Distribution Network for Minimizing Energy Supply Cost by using Multi.Objective BGA", World Academy of Science, Engineering and Technology 45 2008.
- [5] – Shukla, T. N., Singh, S. P., Naik, K. B., "Allocation of optimal distributed generation using GA for minimum system losses in radial distribution networks", International Journal of Engineering , Science and Technology, Vol. 2, No. 3, 2010, pp. 94-106.
- [6] – Liu, B., Yuan, Z., Bao, H., "Intelligent Optimization Algorithm for the locating and Sizing of Distributed Generation Planning".
- [7] – W.F. Tinney, e C.E. Hart, "Power flow solution by Newton's mehod", IEEE Transactions, Vol. PAS-86, 1967.
- [8] – Sucena Paiva, J.P., "Redes de Energia Eléctrica: Uma Análise Sistémica", IST Press, Abril 2005, ISBN: 972-8469-34-9.
- [9] - <http://pt.wikipedia.org/wiki/Quicksort#Python> (2012)

- [10] - <http://www.python.org/> (2012)
- [11] - www.eclipse.org/ (2012)
- [12] – PSS/E 32 Online Documentation – Application Program Interface (API)
- [13] - wxglade.sourceforge.net/ (2012)
- [14] - wxpython.org/ (2012)
- [15] - <http://www.cleargridolutions.com/developer.html> (2012)
- [16] - <http://en.wikipedia.org/wiki/Quicksort>
- [17] – “Plano de Investimentos da Rede Nacional de Transporte 2006-2011”, vol 1, anexo 1 – “Padrões de Segurança de Planeamento da RNT”, pág. 6
- [18] – Wood, A. J., Wollenberg, B. F., “Power Generation, Operation, and Control”
- [19] – Christie, R., “Power system test archive”, Aug. 1999. [Online]. <http://www.ee.washington.edu/research/pstca>

ANEXOS

Anexo A – Dados das redes de teste

Anexo B – Exemplo de código desenvolvido em linguagem Python

ANEXO A

DADOS DAS REDES DE TESTE

Rede de 6 Barramentos

A rede de 6 barramentos foi baseada na rede de teste “IEEE 6 BUS” [18]. O seu esquema unifilar está representado na Figura 30.

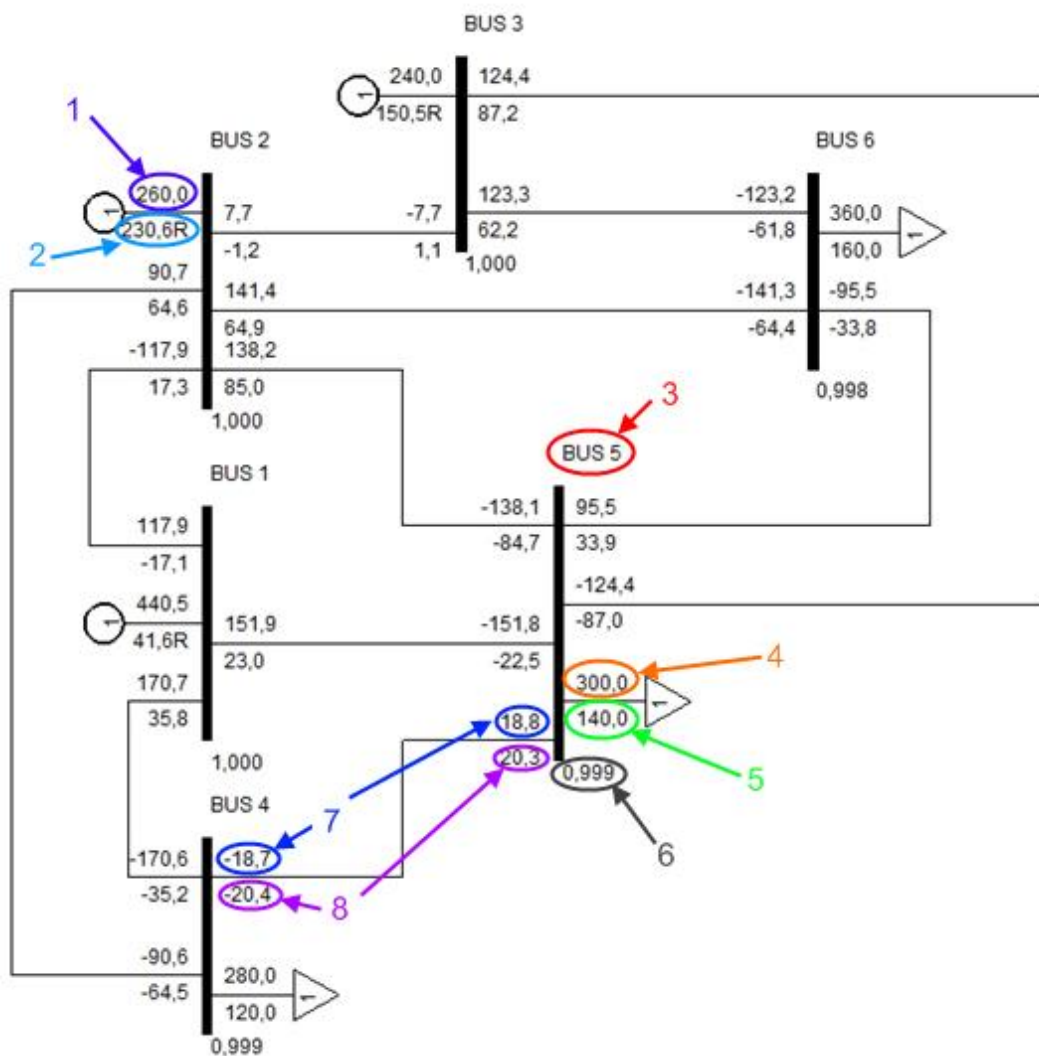


Figura 30 - Rede de 6-barramentos

Legenda:

- 1 – Potência ativa gerada
- 2 – Potência reativa gerada
- 3 – Nome do barramento
- 4 – Potência ativa da carga
- 5 – Potência reativa da carga
- 6 – Tensão no barramento em pu
- 7 - Potência ativa que sai ou entra no barramento
- 8 - Potência reativa que sai ou entra no barramento

Na Tabela 18, estão compilados os dados relativos aos barramentos, geração e carga existente na rede.

Tabela 18 - Dados dos Barramentos (geradores e cargas) da rede de 6 barramentos

Barramento		Tensão	Produção		Carga	
Nº	Tipo	Tensão [pu]	P[MW]	Q[MVar]	P[MW]	Q[MVar]
1	Referência	1,0000	440	41	0	0
2	PV	1,0000	260	230,	0	0
3	PV	1,0000	240	150	0	0
4	PQ	0,9986	0	0	280	120
5	PQ	0,9989	0	0	300	140
6	PQ	0,9981	0	0	360	160

Na Tabela 19, estão representados os parâmetros das linhas que se admitem modelizadas através de um esquema equivalente em π . Os dados em pu estão na base de 100 MVA.

Tabela 19 - Dados das linhas da rede de 6 barramentos

Linhas	Resistência [pu]	Reactância [pu]	Susceptância [pu]
$L_{1\ 2}$	0,000263	0,001799	0,000631
$L_{1\ 4}$	0,000363	0,002199	0,000831
$L_{1\ 5}$	0,000363	0,002299	0,000931
$L_{2\ 3}$	0,000263	0,001799	0,000631
$L_{2\ 4}$	0,000263	0,001799	0,000631
$L_{2\ 5}$	0,000163	0,001000	0,000431
$L_{2\ 6}$	0,000363	0,002199	0,000731
$L_{3\ 5}$	0,000163	0,001000	0,000431
$L_{3\ 6}$	0,000363	0,002399	0,000731
$L_{4\ 5}$	0,000220	0,001400	0,000531
$L_{4\ 6}$	0,000263	0,001799	0,000631

Rede de 14 Barramentos

A rede de 14 barramentos foi baseada na rede de teste “IEEE 14 BUS” [19]. O seu esquema unifilar está representado na Figura 31.

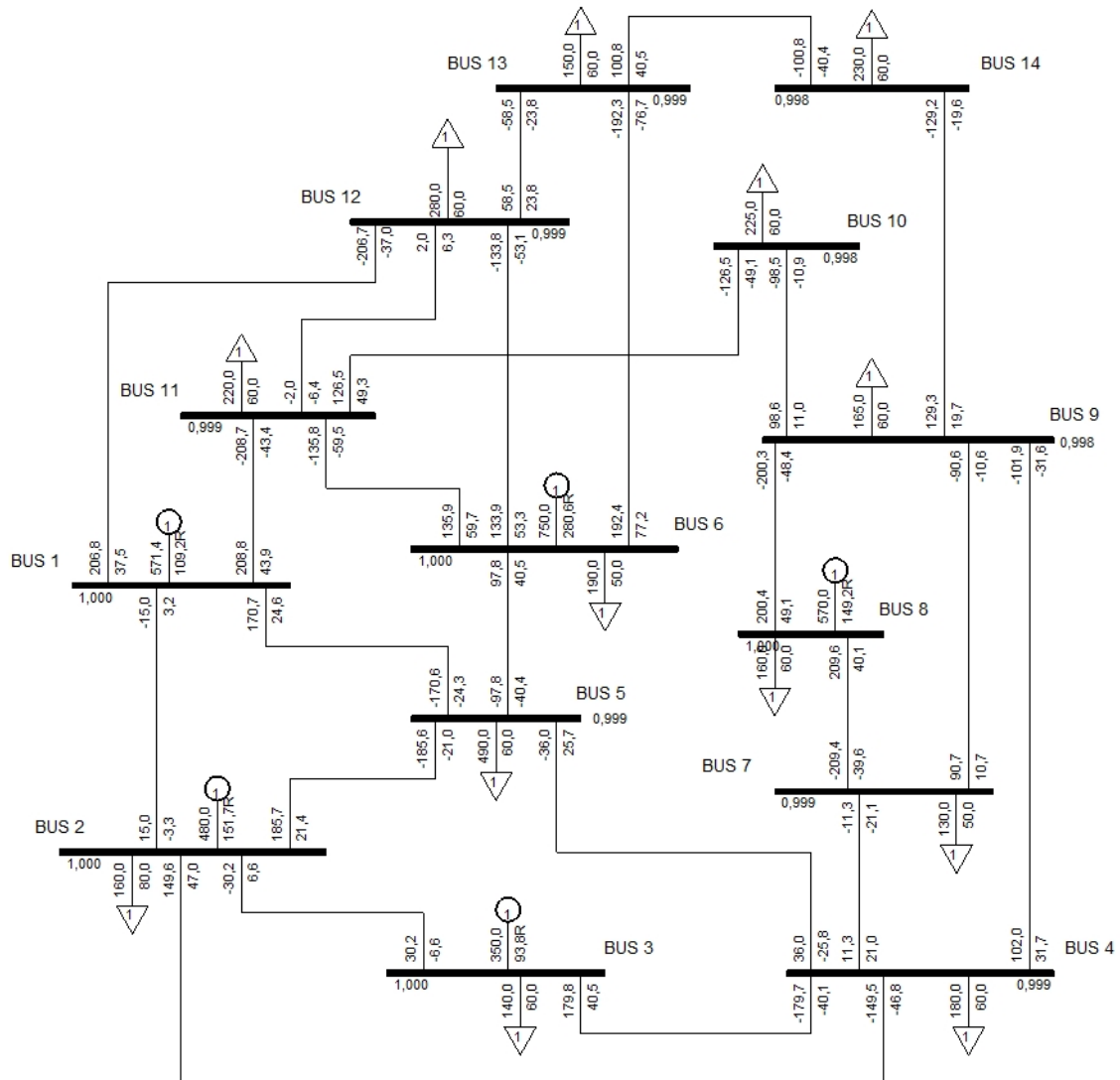


Figura 31 – Rede de 14 barramentos

Na Tabela 20, estão compilados os dados relativos aos barramentos da rede de 14 barramentos.

Tabela 20 - Dados dos Barramentos (geradores e cargas) da rede de 14 barramentos

Barramento		Tensão	Produção		Carga	
Nº	Tipo	Tensão [pu]	P[MW]	Q[MVar]	P[MW]	Q[MVar]
1	Referência	1,0000	571,36	109,17	0	0
2	PV	1,0000	480	151,70	160	80
3	PV	1,0000	350	93,84	140	60
4	PQ	0,9990	0	0	180	60
5	PQ	0,9993	0	0	490	60
6	PV	1,0000	750	280,64	190	50
7	PQ	0,9988	0	0	130	50
8	PV	1,0000	570	149,19	160	60
9	PQ	0,9984	0	0	165	60
10	PQ	0,9980	0	0	225	60
11	PQ	0,9989	0	0	220	60
12	PQ	0,9990	0	0	280	60
13	PQ	0,9986	0	0	150	60
14	PQ	0,9978	0	0	230	60

Na Tabela 21, estão representados os parâmetros das linhas que se admitem modelizadas através de um esquema equivalente em π . Os dados em pu estão na base de 100 MVA.

Tabela 21 - Dados das linhas da rede de 14 barramentos

Linhas	Resistência [pu]	Reactância [pu]	Susceptância [pu]
$L_{1,2}$	0,000263	0,001200	0,000631
$L_{1,5}$	0,000263	0,001200	0,000631
$L_{1,11}$	0,000263	0,001200	0,000631
$L_{1,12}$	0,000263	0,001200	0,000631
$L_{2,3}$	0,000263	0,001200	0,000631
$L_{2,4}$	0,000263	0,001200	0,000631
$L_{2,5}$	0,000263	0,001200	0,000631
$L_{3,4}$	0,000263	0,001200	0,000631
$L_{4,5}$	0,000263	0,001200	0,000631
$L_{4,7}$	0,000263	0,001200	0,000631
$L_{4,9}$	0,000263	0,001200	0,000631
$L_{5,6}$	0,000263	0,001200	0,000631
$L_{6,11}$	0,000263	0,001200	0,000631
$L_{6,12}$	0,000263	0,001200	0,000631
$L_{6,13}$	0,000263	0,001200	0,000631
$L_{7,8}$	0,000363	0,001200	0,000631
$L_{7,9}$	0,000263	0,001200	0,000631
$L_{8,9}$	0,000363	0,001800	0,000631
$L_{9,10}$	0,000263	0,001200	0,000631
$L_{9,14}$	0,000263	0,001200	0,000631
$L_{10,11}$	0,000263	0,001200	0,000631
$L_{11,12}$	0,000263	0,001200	0,000631
$L_{12,13}$	0,000263	0,001200	0,000631
$L_{12,14}$	0,000263	0,001200	0,000631

ANEXO B

Exemplo de código desenvolvido no decurso deste projeto

A título exemplificativo, apresentam-se as partes mais importantes de código de alguns algoritmos.

Algoritmos para o cálculo da potência máxima de injeção nodal não simultânea:

O código seguinte representa grande parte da função “Máxima Injeção Nodal Não simultânea”.

```
def f_individual_Max_Injection(CASE, save_dir, High_PGen = 1000, Accuracy_PGen = 1,
Update_flag = 1):

.....

#Search for best PGen of BUS
for B in Bus_list:

    #The Type of the bus can't be 3 (Swing Bus)
    if B.TYPE <> 3:
        print("Bus Number: ", B.NUMBER)

#=====
# Finding maximum injection Power for bus
#=====

    lower_PG = 0
    high_PG = lower_PG + 20
    Final_PG = 20

    ierr = psspy.case(case_backup)
    while True:

        # Adiciona um gerador no BUS
        Module_1.add_new_generator(B.NUMBER,len(Bus_list)+1, high_PG,230)

        #Solve Power Flow with Newton-Raphson Method
        psspy.fns1([[0,0,0,1,1,0,99,0]])
        temp_1 = psspy.solved() # Check if the last solution reached tolerance
        if temp_1 <> 0:
            print temp_1
            print ("!!!! Erro ao resolver o caso !!!!")

        #Checks if Swing Pgen isn't below threshold (default = 0)
        temp_2 = fGetSwingPower()

        #Verifica se alguma linha chegou ao maximo da sua capacidade
        temp = fGetBranchOverloaded()
```

```

#Hasn't reach branch overload
if temp == 0 and temp_1 == 0 and temp_2 == 1:
    tt = high_PG-lower_PG
    lower_PG = high_PG
    high_PG = lower_PG + tt

#Reached branch overload
else:
    A_temp = lower_PG + (high_PG-lower_PG)/2

    if high_PG-lower_PG > Accuracy_PGen:

        # Adiciona um gerador no BUS
        Module_1.add_new_generator(B.NUMBER,len(Bus_list)+1, A_temp,230)

        #Solve Power Flow with Newton-Raphson Method
        psspy.fns1([[0,0,0,1,1,0,99,0]])
        temp_1 = psspy.solved() # Check solution tolerance
        if temp_1 <> 0:
            print temp_1
            print ("!!!! Erro ao resolver o caso !!!!")

        #Checks if Swing Pgen isn't below threshold (default = 0)
        temp_2 = fGetSwingPower()

        #Verifica se alguma linha chegou ao maximo da sua capacidade
        temp = fGetBranchOverloaded()

        if temp == 0 and temp_1 == 0 and temp_2 == 1:
            lower_PG = A_temp
            Final_PG = A_temp

        else:
            high_PG = A_temp
            Final_PG = lower_PG
    else:
        # Adiciona um gerador no BUS
        Module_1.add_new_generator(B.NUMBER,len(Bus_list)+1, Final_PG,230)

        #Solve Power Flow with Newton-Raphson Method
        psspy.fns1([[0,0,0,1,1,0,99,0]])

        case_a = save_dir + "Bus_" + str(B.NUMBER) + "_" + str(Final_PG) +
"kw" + ".sav"

        if psspy.save(case_a) <> 0:
            print (" Erro na gravacao")

        if temp <> 0:         temp_f = temp
        elif temp_1 <> 0:     temp_f = "Metodo nao convergiu"
        elif temp_2 == 0:    temp_f = "Valor Swing chegou ao Limite minimo"
        else:                 temp_f = "ERRO"

        for B1 in Bus_list:
            #Get Swing bus number
            if B1.TYPE == 3:
                Swing_bus_number = B1.NUMBER
                break

        Reports.fUpdateReportValues(Update_flag, temp_f ,B.NUMBER, Final_PG,
Swing_bus_number)

        print("Reach Bus max: ", case_a)
        break

return 0

```

Algoritmos para o cálculo da potência máxima de injeção nodal simultânea:

A função seguinte representa o algoritmo genético. Nesta função, estão implementadas as várias etapas do algoritmo genético, ver Diagrama 6. Nesta função, são também chamadas outras funções que executam parte do algoritmo genético ou mesmo métodos de ordenação ou injeção nodal.

```
def f_Genetic_Algorithm(CASE, save_dir, High_PGen = 1000, Accuracy_PGen = 1):

    GenerationSize = 40
    PopSize = 30
    EliteSize = 2
    CrossoverRate = 85 #value between 0.1 and 100 (in percentage)
    MutationRate = 8.5 #value between 0.1 and 100 (in percentage)
    PG_min = 0 #min value for initial population generation
    PG_max = 100 #max value for initial population generation
    Accuracy_PGen = 1

    #Number of possible buses (genes) affected by mutation, 0 = all
    MutationOption = 1

    #Current Genetic Algorithm Population
    GA_Population = []

    #Solves the case in questions and saves
    if Module_1.f_solve_case(CASE)<>0:
        print ("!!!! Erro ao resolver o caso !!!!")
        return 1

    case_backup = save_dir + "backup_Genetic_Algorithm" + ".sav"
    if psspy.save(case_backup) <> 0:
        print (" Erro na gravacao")

    Original_Bus_list = Module_1.f_element_info("bus", 0)
    NCromossomas = len(Original_Bus_list) # Number of Cromossoms

    #Generate initial population
    Generate_Population(CASE, GA_Population, Original_Bus_list, PopSize,
    NCromossomas, PG_min, PG_max, Accuracy_PGen, 1)

    for G in range(GenerationSize):

        #=====
        # GET ELITE
        #=====
        #Order Population by fitness
        GA_Population = f_Quick_Sort_List_2(GA_Population)

        #Get list with elites
        Elite_list = []
        for n in range(EliteSize):
            Elite_list.append(GA_Population[len(GA_Population)-1-n])
```

```

#=====

#table to help with the selection, removes sequentially the parents
temp_Pop_list = [n for n in range(PopSize)]

for t_1 in range(PopSize/2):

    #=====
    # Selection
    #=====
    if len(temp_Pop_list) > 2:
        index_1 = random.randint(0,len(temp_Pop_list)-1)
        index_parent_1 = temp_Pop_list[index_1]
        temp_Pop_list.pop(index_1)

        index_2 = random.randint(0,len(temp_Pop_list)-1)
        index_parent_2 = temp_Pop_list[index_2]
        temp_Pop_list.pop(index_2)

    elif len(temp_Pop_list) == 2:
        index_parent_1 = temp_Pop_list[0]
        index_parent_2 = temp_Pop_list[1]
        temp_Pop_list.pop(0)
        temp_Pop_list.pop(0)

    else:
        break

    #=====
    # Crossover
    #=====

    #Checks against CrossoverRate
    if random.randint(1,1000) <= CrossoverRate*10:
        #Get Crossover point
        Crossover_Point = random.randint(1, NCromossomas-1)

        temp_child_1 = []
        temp_child_2 = []

        #
        # print "index_parent_1 = ", index_parent_1
        # print "index_parent_2 = ", index_parent_2
        #
        # print "GA_Population = ", GA_Population

        for nn in range(NCromossomas):
            if nn < Crossover_Point:
                temp_child_1.append(GA_Population[index_parent_1][nn])
                temp_child_2.append(GA_Population[index_parent_2][nn])
            else:
                temp_child_1.append(GA_Population[index_parent_2][nn])
                temp_child_2.append(GA_Population[index_parent_1][nn])

```

```

#=====
# Mutation
#=====
#Checks if the child is valid
if f_Genetic_Validate_Chromosome(CASE, Original_Bus_list,
temp_child_1, NCromossomas) == 1:

    if sum(temp_child_1) <
GA_Population[index_parent_1][NCromossomas]:
        temp_child_1 =
GA_Population[index_parent_1][:NCromossomas]
    else:
        temp_child_1 = GA_Population[index_parent_1][:NCromossomas]

    if MutationOption == 0:
        for n in range(NCromossomas):
            if random.randint(1,1000) <= MutationRate*10:
                #Checks maximum value that bus can inject
                temp_Injection_1 = f_Genetic_Max_Bus_Injection(CASE,
Original_Bus_list, n, temp_child_1, High_PGen = 1000, Accuracy_PGen = 1)
                if temp_Injection_1 > 0 and temp_child_1[n] <
temp_Injection_1:
                    #Puts a random value in bus, between current value
and maximum possible value
                    temp_child_1[n] = random.randint(temp_child_1[n],
temp_Injection_1)
                else:
                    for n1 in range(MutationOption):
                        if random.randint(1,1000) <= MutationRate*10:
                            n = random.randint(0,NCromossomas-1)
                            #Checks maximum value that bus can inject
                            temp_Injection_1 = f_Genetic_Max_Bus_Injection(CASE,
Original_Bus_list, n, temp_child_1, High_PGen = 1000, Accuracy_PGen = 1)
                            if temp_Injection_1 > 0 and temp_child_1[n] <
temp_Injection_1:
                                #Puts a random value in bus, between current value
and maximum possible value
                                temp_child_1[n] = random.randint(temp_child_1[n],
temp_Injection_1)

                    #Adds Fitness
                    temp_child_1.append(sum(temp_child_1))

                    #Substitutes Parent by Children
                    GA_Population[index_parent_1] = temp_child_1

#-----

#-----
#Checks if the child is valid
if f_Genetic_Validate_Chromosome(CASE, Original_Bus_list,
temp_child_2, NCromossomas) == 1:

    if sum(temp_child_2) <
GA_Population[index_parent_2][NCromossomas]:
        temp_child_2 =
GA_Population[index_parent_2][:NCromossomas]
    else:
        temp_child_2 = GA_Population[index_parent_2][:NCromossomas]

```

```

    if MutationOption == 0:
        for n in range(NCromossomas):
            if random.randint(1,1000) <= MutationRate*10:
                #Checks maximum value that bus can inject
                temp_Injection_2 = f_Genetic_Max_Bus_Injection(CASE,
Original_Bus_list, n, temp_child_2, High_PGen = 1000, Accuracy_PGen = 1)
                if temp_Injection_2 > 0 and temp_child_2[n] <
temp_Injection_2:
                    #Puts a random value in bus, between current value
and maximum possible value
                    temp_child_2[n] = random.randint(temp_child_2[n],
temp_Injection_2)
                else:
                    for n1 in range(MutationOption):
                        if random.randint(1,1000) <= MutationRate*10:
                            n = random.randint(0,NCromossomas-1)
                            #Checks maximum value that bus can inject
                            temp_Injection_2 = f_Genetic_Max_Bus_Injection(CASE,
Original_Bus_list, n, temp_child_2, High_PGen = 1000, Accuracy_PGen = 1)
                            if temp_Injection_2 > 0 and temp_child_2[n] <
temp_Injection_2:
                                #Puts a random value in bus, between current value
and maximum possible value
                                temp_child_2[n] = random.randint(temp_child_2[n],
temp_Injection_2)

                                #Adds Fitness
                                temp_child_2.append(sum(temp_child_2))

                                #Substitutes Parent by Children
                                GA_Population[index_parent_2] = temp_child_2

#=====
# Elitism
#=====

# Sees if the best is still the BEST
# Order Population by fitness
GA_Population = f_Quick_Sort_List_2(GA_Population)

#Compare current elites with previous and add them to the population
if they are still better
n_t = 0
for n_1 in range(EliteSize):
    for n_2 in range(EliteSize):
        if Elite_list[n_1][len(Elite_list[0])-1] >
GA_Population[len(GA_Population)-1-n_2][len(GA_Population[0])-1]:
            GA_Population[n_t] = Elite_list[n_1]
            n_t = n_t +1
            break

```

```

#=====
# Saves values for reports
#=====
AverageFitness = 0
for y in GA_Population:
    AverageFitness += y[len(y)-1]

AverageFitness = AverageFitness/len(GA_Population)

#AverageFitness = sum(GA_Population[:,len(GA_Population[0])-1]) /
len(GA_Population)

Reports.lGeneticAlgorithm.append(Reports.cGeneticGeneration(GA_Population[le
n(GA_Population)-1], G, AverageFitness))

#=====
# Save Final Best Case
#=====
ierr = psspy.case(case_backup)

#Create Case of the best individual
for x in range(NCromossomas):

    # Adds a generator at BUS x
    Module_1.add_new_generator(Original_Bus_list[x].NUMBER,
len(Original_Bus_list)+1+x, GA_Population[len(GA_Population)-1][x],230)

#Solve Power Flow with Newton-Raphson Method
psspy.fns1([0,0,0,1,1,0,99,0])
if psspy.solved() <> 0: # Check if the last solution reached tolerance
    print ("!!!! Erro ao resolver o caso !!!!")

case_a = save_dir + "Genetic_best_Individual" + ".sav"
if psspy.save(case_a) <> 0:
    print (" Erro na gravacao")

print GA_Population
print Reports.lGeneticAlgorithm

return 1

```

Funções chamadas pela função principal do algoritmo genético

"f Genetic Algorithm":

A função definida, seguidamente, tem como objetivo gerar a população inicial do algoritmo genético.

```
def Generate_Population(CASE, GA_Population, Bus_list, PopSize,
    NCromossomas, PG_min, PG_max, Accuracy_PGen, flag_first_time = 0):

    n=0
    while True:

        temp_list = [0 for nc in range(NCromossomas+1)]

        #Generates random values for
        for n1 in range(NCromossomas):
            if Bus_list[n1].TYPE <> 3:
                temp_Gen = random.randint(PG_min, PG_max)
                temp_list[n1] = temp_Gen

                # Adiciona um gerador no BUS
                Module_1.add_new_generator(Bus_list[n1].NUMBER,
                    len(Bus_list)+1+n1, temp_Gen,230)

        #Solve Power Flow with Newton-Raphson Method
        psspy.fns1([0,0,0,1,1,0,99,0])
        temp_1 = psspy.solved() #Check if the last solution reached tolerance
        if temp_1 <> 0:
            print temp_1
            print ("!!!! Erro ao resolver o caso !!!!")

        #Checks if Swing Pgen isn't below threshold (default = 0)
        temp_2 = fGetSwingPower()

        #Check if any line is overloaded
        temp = fGetBranchOverloaded()

        #See if individual is valid
        if temp == 0 and temp_1 == 0 and temp_2 == 1:
            #Add fitness result to the individual
            temp_list[len(temp_list)-1] = sum(temp_list)

            #If its the first time, generate the population
            if flag_first_time == 1:
                GA_Population.append(temp_list)
            else:
                GA_Population[n] = temp_list
                n = n+1

        if n > PopSize-1:
            break

    return 1
```

Após a geração da população, é necessário validar os indivíduos ou cromossomas. Esta validação é efetuada pela seguinte função.

```
def f_Genetic_Validate_Chromosome(CASE_t, Bus_list, cromosoma, NCromossomas):  
    ierr = psspy.case(CASE_t)  
  
    #Generates random values for chromosomes  
    for n1 in range(NCromossomas):  
        if Bus_list[n1].TYPE <> 3:  
            # Adiciona um gerador no BUS  
            Module_1.add_new_generator(Bus_list[n1].NUMBER, len(Bus_list)+1+n1,  
cromosoma[n1],230)  
  
            #Solve Power Flow with Newton-Raphson Method  
            psspy.fns1([0,0,0,1,1,0,99,0])  
            temp_1 = psspy.solved() # Check if the last solution reached tolerance  
            if temp_1 <> 0:  
                print temp_1  
                print ("!!!! Erro ao resolver o caso !!!!")  
  
            #Checks if Swing Pgen isn't below threshold (default = 0)  
            temp_2 = fGetSwingPower()  
  
            # Checks if any branch reached max threshold  
            temp = fGetBranchOverloaded()  
  
            #See if individual is valid  
            if temp == 0 and temp_1 == 0 and temp_2 == 1:  
                #Add fitness result to the individual  
                #cromosoma.append(sum(cromosoma))  
  
                return 1  
  
    return -1
```

O código seguinte representa o algoritmo *Quicksort*, de forma a ordenar uma tabela bidimensional. É usado no método Heurístico e segue um processo iterativo e recursivo, chamando a própria função onde está inserido.

```
#algoritmo quicksort
def qsort1(list):
    if list == []:
        return []
    else:
        pivot = list[0]
        lesser = qsort1([x for x in list[1:] if x < pivot])
        greater = qsort1([x for x in list[1:] if x >= pivot])
        return lesser + [pivot] + greater
```