



Universidade Federal de Campina Grande

Centro de Engenharia Elétrica e Informática

Curso de Graduação em Engenharia Elétrica

MARTINS MARQUES BORBUREMA NETO

**UMA ADAPTAÇÃO DO MSP PARA ANÁLISE DE REDES COM
GERAÇÃO DISTRIBUÍDA**

Campina Grande, Paraíba
Outubro de 2014

MARTINS MARQUES BORBUREMA NETO

UMA ADAPTAÇÃO DO MSP PARA ANÁLISE DE REDES COM
GERAÇÃO DISTRIBUÍDA

*Trabalho de Conclusão de Curso submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Sistemas de Potência

Orientador:

Professor Benemar Alencar Souza, D. Sc.

Campina Grande, Paraíba
Outubro de 2014

MARTINS MARQUES BORBUREMA NETO

UMA ADAPTAÇÃO DO MSP PARA ANÁLISE DE COM GERAÇÃO DISTRIBUÍDA

*Trabalho de Conclusão de Curso submetido à Unidade
Acadêmica de Engenharia Elétrica da Universidade
Federal de Campina Grande como parte dos requisitos
necessários para a obtenção do grau de Bacharel em
Ciências no Domínio da Engenharia Elétrica.*

Área de Concentração: Sistemas de Potência

Aprovado em 09 / 10 /2014

Professor Convidado
Universidade Federal de Campina Grande
Avaliador

Professor Benemar Alencar Souza, D. Sc.
Universidade Federal de Campina Grande
Orientador, UFCG

Dedico este trabalho a meus pais, Edilson e Bernadete, cujo esforço e dedicação fizeram tudo possível.

AGRADECIMENTOS

Agradeço a meus pais, Edilson e Bernadete, sem seu esforço e comprometimento jamais teria a boa educação necessária para chegar até aqui.

Agradeço a Anna Carolina, fiel companheira de todas as horas, com quem dividi as alegrias e tristezas dessa etapa de minha vida.

Agradeço também à minha família, que me deu todo o suporte e carinho necessários para me ajudar e ajudar meus pais nessa longa caminhada.

Agradeço aos amigos de infância, aos que fiz na universidade e aos que fiz no intercâmbio, que me ajudaram, mesmo sem saber, seja com sua companhia nas noites de estudo, com suas sábias palavras de conforto e um ombro amigo nas horas difíceis.

Agradeço ao meu orientador, Benemar Alencar, pela paciência e tempo dedicado.

Agradeço também ao professor Maurício pela primeira oportunidade que tive como pesquisador na UFCG.

Enfim, agradeço a todos que de alguma forma, passaram pela minha vida e contribuíram para a construção de quem sou hoje.

“Sonhos determinam o que você quer. Ação determina o que você conquista.”

Aldo Novak.

RESUMO

O cálculo do fluxo de carga sempre foi algo muito importante no estudo das redes de distribuição, para saber como a mesma se comporta para determinadas condições de cargas, topologias, geração, entre outros.

Ao longo dos anos vários métodos foram criados para cálculo do fluxo de potência, dentre eles podemos destacar o MSP (Método de Soma de Potência) que se utiliza de determinadas características das redes de distribuição para ser um método bastante eficiente e de rápida convergência.

O MSP aproveitava o fato de todas as barras da rede serem de carga para ser mais rápido, mas como sabemos atualmente a realidade é outra.

Este trabalho tem como objetivo adaptar o Método de Soma de Potência, usando métodos de otimização, para que o mesmo possa ser usado em cálculo de fluxo de potência em redes de geração distribuída e assim continuar se aproveitando das vantagens que o método apresenta.

Palavras-chave: MSP, Geração Distribuída, Fluxo de Carga, Redes de Distribuição

SUMÁRIO

1	Introdução.....	1
1.1	Objetivo	1
1.2	Metodologia	2
1.3	Estrutura do Trabalho	2
2	Formulação Matemática	2
3	Algoritmo da Solução'	7
3.1	MSP original	7
3.2	Alterações no MSP	10
4	Resultados	12
4.1	Alimentador de treze barras	12
4.2	Alimentador de trinta e seis barras	13
5	Conclusão	16
	Bibliografia.....	17
	APÊNDICE A	18

1 INTRODUÇÃO

O Método da Soma de Potências (MSP) é um método de cálculo de fluxo de carga iterativo que calcula as variáveis perdas de potências ativa e reativa, e é do tipo *forward-backward*, ou seja, ele anda em duas direções em cada iteração.

Com os dados iniciais, incluindo a tensão da barra da subestação e as perdas sendo consideradas nulas na primeira iteração, se calcula a tensão em todas as barras ligadas diretamente à subestação, depois nas barras ligadas a essas e assim por diante, respeitando os níveis de cada uma.

Ao fim desse primeiro estágio, que é chamado de *forward*, têm-se os valores aproximados de todas as tensões de barra. Com esses valores começamos o estágio *backward* do MSP, onde se calculam as perdas em todos os trechos e então corrigimos os fluxos de potência com os novos valores. O processo completo (*forward-backward*) continua enquanto a variação nas perdas totais for maior que uma tolerância previamente escolhida ou se eventualmente o limite de iterações for excedido.

A eficiência do MSP se deve a ele explorar as características do alimentador: configuração radial, condutância e susceptância em derivação desprezíveis e todas as barras serem de carga, exceto a da subestação, que é a barra de balanço. Até o passado recente era exatamente isso que acontecia, mas ultimamente, com a geração distribuída, há possibilidade de algumas barras serem de geração. (SOUZA, 2005)

Diferentemente de uma barra de carga, na qual são especificadas as potências ativa e reativa e se precisa calcular a tensão, numa barra de geração são especificadas a tensão e a potência ativa, e se precisa calcular a potência reativa. Para resolver o novo problema, será utilizado a *toolbox* de otimização do *Matlab*[®].

1.1 OBJETIVO

Adaptar o Método da Soma de Potências para resolução e análise de sistemas de distribuição com geração distribuída, usando métodos de otimização disponíveis no *software Matlab*[®].

1.2 METODOLOGIA

O Trabalho foi executado conforme as atividades apresentadas a seguir:

1. Estudo e implementação do método usando um sistema de distribuição simples;
2. Adaptação do código para uso em problemas mais complexos;
3. Uso do *toolbox* de otimização do *Software Matlab*[®] para adaptar o MSP para simulação de redes de distribuição com geração distribuída;
4. Confecção do relatório.

Para o desenvolvimento do Trabalho, foram utilizados os seguintes recursos didáticos:

- *Software Matlab*[®];

1.3 ESTRUTURA DO TRABALHO

Este trabalho está organizado em cinco capítulos. No capítulo 1, uma introdução do tema é feita e também é apresentada a metodologia usada no trabalho e seu objetivo.

No capítulo 2 é realizada a formulação matemática do problema do MSP, desenvolvimento necessário para a correta implementação do problema.

No capítulo 3, há uma explicação de como o algoritmo é implementado e quais alterações foram necessárias para adaptá-lo ao novo paradigma.

No capítulo 4 é mostrado o resultado da implementação usando duas redes com geração distribuída como exemplo

O capítulo 5 apresenta as conclusões que foram obtidas, mostrando os aspectos positivos da implementação e a contribuição que o trabalho dá no âmbito do cálculo de fluxo de carga.

2 FORMULAÇÃO MATEMÁTICA

Tomando como exemplo um sistema radial com um trecho da seguinte maneira:

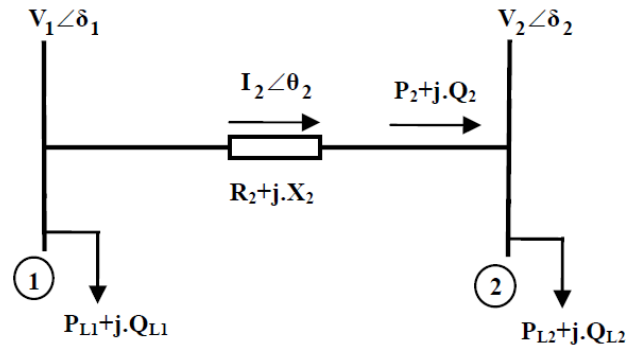


Figura 1: Modelo de Trecho de Rede de Distribuição (Fonte: ALBUQUERQUE,2005)

A Partir da Figura 1, duas equações podem ser estabelecidas:

$$I_2 = \frac{V_1 \angle \delta_1 - V_2 \angle \delta_2}{R_2 + j \cdot X_2} \quad (1)$$

$$S_2 = V_2 \cdot I_2^* \rightarrow S_2^* = V_2^* \cdot I_2 \rightarrow P_2 - j \cdot Q_2 = V_2^* \cdot I_2$$

$$I_2 = \frac{(P_2 - j \cdot Q_2)}{V_2^*} \quad (2)$$

Igualando as equações (1) e (2) tem-se:

$$\frac{(P_2 - j \cdot Q_2)}{V_2^*} = \frac{V_1 \angle \delta_1 - V_2 \angle \delta_2}{R_2 + j \cdot X_2}$$

$$(V_2 \angle -\delta_2) \cdot (V_1 \angle \delta_1 - V_2 \angle \delta_2) = (P_2 - j \cdot Q_2) \cdot (R_2 + j \cdot X_2)$$

$$V_1 \cdot V_2 \cdot \angle(\delta_1 - \delta_2) - V_2^2 = P_2 \cdot R_2 + j \cdot P_2 \cdot X_2 - j \cdot Q_2 \cdot R_2 + Q_2 \cdot X_2$$

$$\begin{aligned} V_1 \cdot V_2 \cdot [\cos(\delta_1 - \delta_2) + j \cdot \text{sen}(\delta_1 - \delta_2)] \\ = [(V_2^2 + P_2 \cdot R_2 + Q_2 \cdot X_2) + j \cdot (P_2 \cdot X_2 - Q_2 \cdot R_2)] \end{aligned}$$

Separando as partes real e imaginária, tem-se:

$$V_1 \cdot V_2 \cdot \cos(\delta_1 - \delta_2) = V_2^2 + P_2 \cdot R_2 + Q_2 \cdot X_2 \quad (3)$$

$$V_1 \cdot V_2 \cdot \text{sen}(\delta_1 - \delta_2) = P_2 \cdot X_2 - Q_2 \cdot R_2 \quad (4)$$

Elevando-se ao quadrado e somando-se as equações (3) e (4), tem-se:

$$\begin{aligned} V_1^2 \cdot V_2^2 \cdot [\cos^2(\delta_1 - \delta_2) + \text{sen}^2(\delta_1 - \delta_2)] \\ = (V_2^2 + P_2 \cdot R_2 + Q_2 \cdot X_2)^2 + (P_2 \cdot X_2 - Q_2 \cdot R_2)^2 \end{aligned}$$

$$V_1^2 \cdot V_2^2 = V_2^4 + 2 \cdot V_2^2 \cdot (R_2 \cdot P_2 + X_2 \cdot Q_2) + (R_2 \cdot P_2 + X_2 \cdot Q_2)^2 + (X_2 \cdot P_2 - R_2 \cdot Q_2)^2$$

$$\begin{aligned} V_2^4 + 2 \left[(R_2 \cdot P_2 + X_2 \cdot Q_2) - \frac{1}{2} \cdot V_1^2 \right] V_2^2 \\ + [(P_2^2 \cdot R_2^2 + 2 \cdot R_2 \cdot P_2 \cdot X_2 \cdot Q_2 + Q_2^2 \cdot X_2^2) \\ + (P_2^2 \cdot X_2^2 - 2 \cdot R_2 \cdot P_2 \cdot X_2 \cdot Q_2 + Q_2^2 \cdot R_2^2)] = 0 \end{aligned}$$

$$V_2^4 + 2 \left[(R_2 \cdot P_2 + X_2 \cdot Q_2) - \frac{1}{2} \cdot V_1^2 \right] V_2^2 + (R_2^2 + X_2^2) \cdot (P_2^2 + Q_2^2) = 0$$

A igualdade apresentada em cima pode ser reescrita da seguinte forma:

$$V_2^4 + 2 \cdot A \cdot V_2^2 + B = 0 \quad (5)$$

Onde:

$$A = (R_2 \cdot P_2 + X_2 \cdot Q_2) - \frac{1}{2} \cdot V_1^2 \quad (6)$$

$$B = (R_2^2 + X_2^2) \cdot (P_2^2 + Q_2^2) \quad (7)$$

Como podemos ver, a equação (5) não depende da fase das tensões, o que facilita bastante a resolução do problema. No geral, em sistemas de distribuição, as fases das tensões não são fatores de extrema importância, uma vez que a diferença de fase entre o nó da subestação e o último nó do alimentador normalmente é de alguns graus.

Vale a pena ressaltar que a equação biquadrada possui quatro raízes. Entretanto, das duas soluções para V_2^2 , porém apenas a solução que considera o sinal positivo da raiz quadrada da solução fornece um valor de tensão possível. Depois há mais duas soluções para as raízes de V_2 e mais uma vez considera-se que apenas a solução com raiz de sinal positivo fornece uma tensão possível.

A partir destas conclusões obtém-se a solução da equação da seguinte maneira:

$$\begin{aligned}
V_2^4 + 2.A.V_2^2 + B &= 0 \\
\gamma = V_2^2 \rightarrow \gamma^2 + 2.\gamma.A + B &= 0 \\
\gamma = \frac{-2.A \mp \sqrt{4.A^2 - 4.B}}{2} &= \frac{-2.A \mp 2.\sqrt{A^2 - B}}{2} = -A \mp \sqrt{A^2 - B} \\
\gamma &= \sqrt{A^2 - B} - A \\
V_2 = \mp \sqrt{\gamma} \rightarrow V_2 &= \sqrt{\sqrt{A^2 - B} - A} \tag{8}
\end{aligned}$$

Após a determinação da solução possível da equação biquadrada que fornece o valor do módulo das tensões nos diversos nós do alimentador, podemos chegar a importantes conclusões. A primeira delas é que a tensão em um determinado trecho é função dos parâmetros da linha daquele trecho, do fluxo de potência no mesmo, como também da tensão no trecho do qual se origina o trecho em questão.

Apesar de toda a demonstração ter sido usada com um alimentador em específico, nada impede que a formulação matemática acima seja usada em todos os trechos. Dessa forma podemos transformar as expressões para um determinado trecho genérico i do alimentador:

$$V_i^4 + 2.A.V_i^2 + B = 0 \tag{9}$$

$$A = (R_i.P_i + X_i.Q_i) - \frac{1}{2}.V_{i-l}^2 \tag{10}$$

$$B = (R_i^2 + X_i^2).(P_i^2 + Q_i^2) \tag{11}$$

Como agora temos todas as tensões nos nós do alimentador disponíveis, o próximo passo é calcular as perdas ativa e reativa em cada trecho utilizando as seguintes equações:

$$\begin{aligned}
\Delta P_i = R_i.I_i^2 \rightarrow \Delta P_i &= R_i.\left(\frac{S_i}{V_i}\right)^2 \\
\Delta P_i = R_i.\left(\frac{P_i^2 + Q_i^2}{V_i^2}\right)^2 & \tag{12} \\
\Delta Q_i = X_i.I_i^2 \rightarrow \Delta Q_i &= X_i.\left(\frac{S_i}{V_i}\right)^2
\end{aligned}$$

$$\Delta Q_i = X_i \cdot \left(\frac{P_i^2 + Q_i^2}{V_i^2} \right)^2 \quad (13)$$

Segundo as equações, as perdas em cada trecho dependem, além do módulo da tensão no trecho em estudo, do fluxo de potência no mesmo. Ainda usando a mesma figura como exemplo, mas a tratando como um trecho genérico i , determina-se os fluxos de potência ativa e reativa utilizando-se as expressões (SOUZA, 2004):

$$P_i = P_{Li} + \sum_{k \in \Omega_i} (P_k + \Delta P_k) \quad (14)$$

$$Q_i = Q_{Li} + \sum_{k \in \Omega_i} (Q_k + \Delta Q_k) \quad (15)$$

Onde:

- P_i e Q_i são o fluxo de potência ativa e reativa no trecho i ;
- P_{Li} e Q_{Li} são as cargas ativa e reativa instaladas no trecho i ;
- ΔP_k e ΔQ_k são as perdas ativa e reativa no trecho k .

Nestas equações, Ω_i é o conjunto de todos os trechos que derivam do trecho i . Logo podemos dizer que o fluxo de potência em um determinado trecho i é dado pela carga instalada no mesmo, somado do fluxo de potência e das perdas das barras que derivam do mesmo.

Por último, é possível desenvolver uma expressão para o cálculo das fases das tensões nas barras do alimentador, embora esta não seja uma informação comumente desejada. Assim, temos o seguinte desenvolvimento matemático:

$$V_1 \cdot V_2 \cdot \text{sen}(\delta_1 - \delta_2) = P_2 \cdot X_2 - Q_2 \cdot R_2$$

$$\text{sen}(\delta_1 - \delta_2) = \frac{P_2 \cdot X_2 - Q_2 \cdot R_2}{V_1 \cdot V_2}$$

Assumindo $\delta_1 - \delta_2 = 0$:

$$\delta_2 = \delta_1 - \frac{P_2 \cdot X_2 - Q_2 \cdot R_2}{V_1 \cdot V_2}$$

Ou, de forma genérica:

$$\delta_i = \delta_{i-1} - \frac{P_i \cdot X_i - Q_i \cdot R_i}{V_{i-1} \cdot V_i} \quad (16)$$

Através da análise da equação (16) é possível verificar que a fase da tensão em uma barra depende tanto do módulo da tensão nesta barra quanto do módulo e da fase da tensão na barra de origem. Portanto, o cálculo das fases das tensões de barra pode ser feito à medida que os módulos das tensões são calculados, na direção da subestação para as barras terminais.

Como já sabemos a tensão na subestação e sua fase é considerada nula, podemos usá-la este local como ponto de partida para o cálculo das demais fases das tensões de barra.

3 ALGORITMO DA SOLUÇÃO

A seguir será descrito o procedimento que foi usado para implementar a resolução do problema de fluxo de carga em redes de distribuição. Primeiro será falado sobre como foi implementado o MSP e depois as alterações feitas para adaptá-lo para o uso em problemas com geração distribuída.

3.1 MSP ORIGINAL

Inicialmente se criou um arquivo no formato .xls do Microsoft® Office Excel contendo todos os dados de entrada do programa, que são as informações básicas sobre cada trecho do alimentador:

	A	B	C	D	E
1	Tensão e Qtde de Barras de Geração	13,8	2	0	
2	Início do Trecho	0	1	1	
3	Fim do Trecho	1	2	3	
4	R	3	4	4	
5	X	4	5	3	
6	P	0,64	0,8	0,32	
7	Q	0,48	0,6	0,24	
8	Barras de Geração	2	3		
9	Potência Gerada (Seguir ordem da barra)	1,2	1		
10	Potência Reativa Mínima (Seguir Ordem da Barra)	0,5812	0,4843		
11	Potência Reativa Máximo (Seguir Ordem da Barra)	0,9	0,75		
12	Tensão Controlada (Seguir ordem da barra)	13,7	13,7		
13					

Figura 2: Arquivo de Entrada

Os dados são:

- Tensão na barra da subestação;
- Quantidade de Barras de Geração;
- Nó de origem;
- Nó de destino;
- Resistência da linha;
- Reatância da linha;
- Carga ativa instalada;
- Carga reativa instalada;
- Potência ativa gerada;
- Potência reativa mínima;
- Potência reativa máxima;
- Valor de tensão que se espera na barra de geração.

É importante destacar que nem todas as variáveis recebidas são para uso no MSP original. Depois de receber os dados do programa, o primeiro passo foi determinar o nível e o tipo de cada barra. Para isso dividimos as barras em três tipos, descritos na tabela a seguir:

Tabela I: Tipos de Barra

Tipo de Barra	Codificação	Descrição
Terminal	0	É a barra onde se termina um tronco, ramal ou um lateral.
Intermediária	1	A barra da onde deriva apenas uma barra
Derivação	2	A barra da onde deriva mais de uma barra

Essa determinação se faz necessária porque o cálculo do fluxo de carga em cada trecho difere para cada tipo de barra. Para barras terminais leva-se em consideração apenas a carga instalada naquele trecho, já para barras intermediárias e de derivação leva-se em consideração a carga instalada naquele trecho somado das cargas e das perdas dos trechos que derivam daquela barra. É importante frisar que as perdas não são

levadas em consideração no trecho em que foram calculadas e sim no trecho anterior, pois quando calculamos as perdas, elas são calculadas para o fim de cada trecho. Logo após a determinação do tipo de cada barra, vamos à definição do nível de cada barra. O Nível é um conceito bastante importante no cálculo de fluxo de carga, pois ela evita eventuais erros de cálculo na implementação do algoritmo.

O Nível dos trechos que partem da subestação é o nível 1, os trechos que partem dos trechos de nível 1 fazem parte do nível 2 e assim sequencialmente e em ordem crescente até que as barras terminais mais distantes da subestação tenham sido classificadas.

Como já foi dito anteriormente o MSP é um método *forward-backward*, ou seja, a cada nova iteração os cálculos seguem duas direções diferentes. Na parte *backward* do processo iterativo o fluxo de potência é calculado com o auxílio das equações (14) e (15) Esse cálculo começa das barras terminais e de maior nível, onde o fluxo de carga é representado apenas pelas cargas instaladas naquele trecho e vai seguindo até chegar às barras de nível 1.

Em seguida, com o fluxo de potência já determinado, agora se dá início à etapa *forward* do MSP. Com os parâmetros da linha já dados inicialmente e o fluxo de potência usamos as equações (9), (10) e (11) para calcular as tensões nas barras do alimentador. Como a tensão de uma barra depende da tensão da barra anterior, nessa etapa começamos os cálculos a partir das barras de nível 1, pois as de nível 0 já têm sua tensão determinada e assim vamos seguindo até chegar às barras terminais de maior nível.

Depois de finalizado os cálculos das tensões, usamos as equações (12) e (13) para atualizar os valores das perdas e assim encerrar a iteração.

Porém há um pequeno problema durante a primeira iteração. Para calcular as perdas, precisamos das tensões e do fluxo de potência, porém não podemos obter o fluxo de potência porque ainda não temos os valores das perdas. Para contornar esse problema consideramos na primeira iteração os valores das perdas como sendo nulos, para que depois da primeira iteração já tenhamos um valor aproximado e assim os cálculos possam seguir tranquilamente até que a convergência seja alcançada. Depois de explicado todo o algoritmo pode ser simplificado nas seguintes etapas:

1. Antes da primeira iteração calcula-se o nível e o tipo de cada barra;

2. Para a primeira iteração considerar nulas as perdas ativa e reativa no alimentador;
3. Calcular os fluxos em cada trecho utilizando as equações (14) e (15), partindo das barras de maior nível para as de menor nível;
4. Calcular a tensão em cada trecho usando as equações (9), (10) e (11), partindo das barras da subestação até as barras de maior nível;
5. Calcular as perdas ativa e reativa em cada trecho usando as equações (12) e (13);
6. Repetir os passos 3 a 5 até que não haja uma variável menor que a tolerância estabelecida entre as perdas totais de uma iteração e outra.

3.2 ALTERAÇÕES NO MSP

Para que se entenda o porquê das alterações estarem sendo feitas, primeiro é necessário entender o que realmente é o problema, que é saber se é possível que a barra mantenha uma tensão desejada através da injeção de uma quantidade de reativos que têm que estar em uma faixa pré-determinada, pois todos os geradores possuem limitações da quantidade de potência gerada.

Uma maneira de se fazer isso é através de um problema de otimização, onde uma variável erro, que é função da potência reativa injetada na barra, representada pela soma das diferenças entre a tensão na barra e a tensão desejada para a barra tem que ser minimizada.

Quando há n barras de geração, podemos descrever o problema da seguinte maneira:

Minimizar

$$f(Q_{\phi_1}, Q_{\phi_2}, \dots, Q_{\phi_n}) = \sum_{i \in \Phi} (V_i - \bar{V}_i)^2$$

sujeito a

$$Q_{\min_{\phi_i}} \leq Q_{\phi_i} \leq Q_{\max_{\phi_i}}$$

Sendo $\Phi = \{\phi_1, \phi_2, \dots, \phi_n\}$ o conjunto de barras de geração.

Existem vários métodos de resolução desse problema de otimização, e optamos pelo MATLAB ©, mais especificamente a função *fmincon*.

Essa função foi escolhida por ser bastante eficiente e abrangente, podendo resolver vários problemas, incluindo o proposto, pois a função implementa a restrição de domínio das variáveis de controle, que é algo de extremamente importância, pois como já foi dito, geradores apresentam limitações de quantidade de reativos que podem ser gerados.

Utilizamos o algoritmo *interior point* já implementado na função *fmincon* para realizar a otimização.

fmincon resolve o problema de otimização da seguinte maneira:

$$\min_x f(x)$$

sujeito a

$$A \cdot x \leq b \quad (17)$$

$$A_{eq} \cdot x = b_{eq} \quad (18)$$

$$C(x) \leq 0 \quad (19)$$

$$C_{eq} \cdot x = 0 \quad (20)$$

$$b_L \leq x \leq b_U \quad (21)$$

Onde (17) e (18) são as restrições lineares, (19) e (20) as restrições não-lineares e (21) é a restrição das variáveis de controle. A forma como implementamos a função *fmincon* é a seguinte:

$$x = \text{fmincon}(\text{fun}, x0, A, B, Aeq, Beq, lb, ub)$$

É importante destacar que *fun* é uma função escalar, ou seja, retorna um único valor escalar quando a mesma é chamada.

No problema em questão, não há restrições lineares e nem não-lineares, apenas as restrições de domínio. Logo a chamada da função deve ser feita da seguinte maneira:

$$QGotimo = \text{fmincon}(@delV, QG0, [], [], [], [], QGmin, QGmax)$$

Onde *QGmin* e *QGmax* são vetores que representam as potências reativas geradas máximas e mínimas para cada barra. *QG0* é uma estimativa inicial das potências reativas geradas e pode ser dado como sendo a média entre os limites máximo

e mínimo da potência reativa gerada. delV é uma função que contém o algoritmo do MSP e quando chamada retorna a variável erro anteriormente descrita.

4 RESULTADOS

Para comprovação de funcionamento do algoritmo foram utilizados dois alimentadores com alguns geradores instalados nos mesmos.

4.1 ALIMENTADOR DE TREZE BARRAS

Este sistema foi retirado de ALBUQUERQUE, (2005), com apenas uma leve alteração no comprimento de alguns trechos. A alteração foi feita, pois com os dados originais estava sendo realizada uma divisão de zero por zero, gerando assim um erro em cascata que prejudicava todo o resultado final, por isso foi optado usar distância muito pequena para resolver o problema. Na figura 3 a seguir é mostrado o diagrama esquemático do alimentador. Na tabela II temos os dados de todos os trechos do alimentador e na tabela III as informações dos geradores.

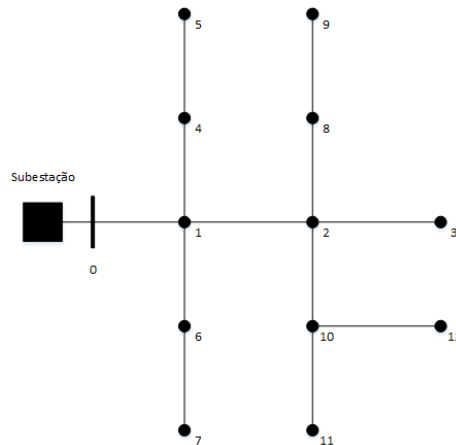


Figura 3: Diagrama esquemático de alimentador de treze barras

Tabela II: Dados de Entrada e de Saída do alimentador de treze barras

<i>De</i>	<i>Para</i>	$R(\Omega)$	$X(\Omega)$	$P_l(MW)$	$Q_l(Mvar)$	$V(kV)$	$P(MW)$	$Q(Mvar)$
0	1	0,070455	0,15951	0,06667	0,03867	4,138605	0,529016	0,320739
1	2	0,070455	0,15951	0	0	4,122898	0,394604	0,2313
2	3	0,035227	0,079755	0	0	4,122898	0	0
1	4	0,056061	0,055019	0	0	4,14	-0,06667	-0,03699

4	5	5,07E-07	9,22E-07	0,13333	0,09667	4,14	0,13333	0,09667
1	6	0,106061	0,062121	0,05667	0,04167	4,133896	0,133369	0,085687
6	7	0,063636	0,037273	0,07667	0,044	4,132318	0,07667	0,044
2	8	1E-13	1E-13	0,05667	0,05033	4,122898	0,337905	0,204613
8	9	0,038826	0,04675	0,281	0,154	4,1185	0,281	0,154
2	10	0,063636	0,037273	0	0	4,121782	0,056685	0,026679
10	11	0,063636	0,037273	0,05667	0,02667	4,120665	0,05667	0,02667
10	12	0,169697	0,099394	0	0	4,121782	0	0

Tabela III: Dados do Gerador Instalado no alimentador de treze barras

<i>Barra</i>	$P_G(MW)$	fp_{min}	fp_{max}	$V_{min}(kV)$	$\bar{V}(kV)$	$V_{max}(kV)$
4	0,2	0,8	0,9	4,1325	4,14	4,14

O algoritmo conseguiu chegar a um resultado ótimo de potência reativa que permitiu manter a tensão controlada no valor que se queria. O resultado é apresentado na tabela a seguir:

Tabela IV: Resultados do Gerador

<i>Barra</i>	$Q_{min}(Mvar)$	$Q(Mvar)$	$Q_{max}(Mvar)$	$\bar{V}(kV)$	$V(kV)$
4	0,0968	0,1337	0,15	4,14	4,14

4.2 ALIMENTADOR DE TRINTA E SEIS BARRAS

Este sistema foi retirado de SOUZA,(2006) sem nenhuma modificação. Na figura 4 a seguir é mostrado o diagrama esquemático do alimentador. Na tabela V temos os dados de todos os trechos do alimentador e na tabela VI as informações dos geradores.

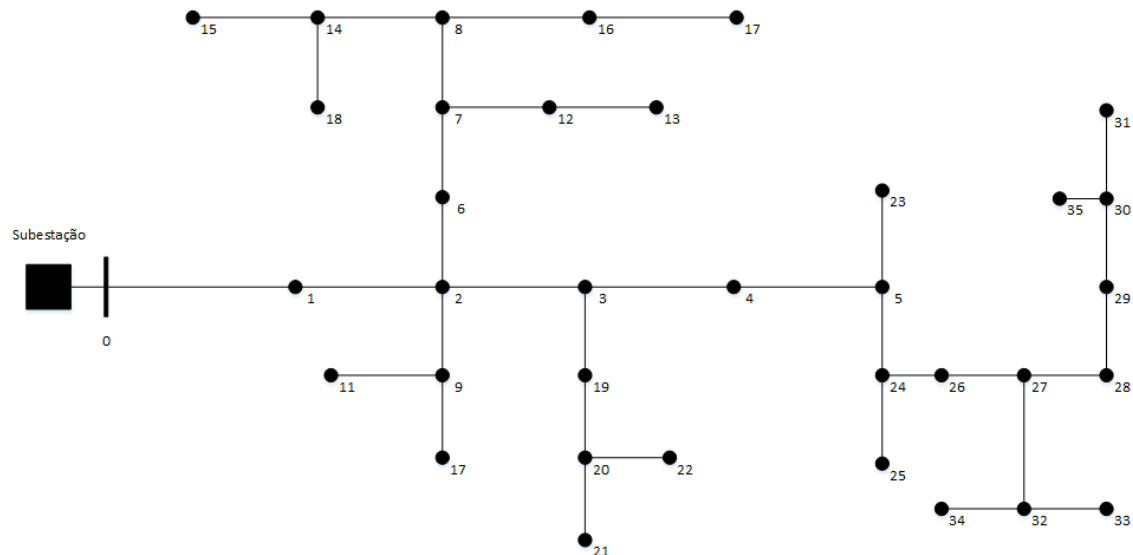


Figura 4: Diagrama esquemático de alimentador de 36 barras

Tabela V: Dados de Entrada e Saída do Alimentador de trinta e seis barras

<i>De</i>	<i>Para</i>	$R(\Omega)$	$X(\Omega)$	$P_l(MW)$	$Q_l(Mvar)$	$V(kV)$	$P(MW)$	$Q(Mvar)$
0	1	0,0368	0,1404	0,84	0,42	4,75887	1,844434	0,902723
1	2	0,0375	0,0805	0	0	4,742808	1,002378	0,478308
2	3	0,0515	0,1107	0	0	4,728699	0,668147	0,291152
3	4	0,1017	0,0728	0,1133	0,0533	4,694867	1,158161	0,563639
4	5	0,0339	0,0243	0	0	4,684681	1,042782	0,508848
2	6	0,061	0,0437	0,1133	0,0533	4,740916	0,095182	0,072317
6	7	0,0881	0,0631	0	0	4,741	-0,01812	0,019015
7	8	0,1356	0,0971	0,1133	0,0533	4,723883	0,441271	0,21648
2	9	0,1167	0,0435	0	0	4,735928	0,237427	0,112047
9	10	0,0933	0,0348	0,124	0,0587	4,733052	0,124	0,0587
9	11	0,07	0,0261	0,1133	0,0533	4,733959	0,1133	0,0533
7	12	0,0233	0,0087	0,0507	0,024	4,740051	0,164106	0,077339
12	13	0,1517	0,0565	0,1133	0,0533	4,735786	0,1133	0,0533
8	14	0,2683	0,1	0	0	4,705577	0,27083	0,134748
14	15	0,2217	0,0826	0,056	0,028	4,702445	0,056	0,028
8	16	0,1017	0,0728	0	0	4,722244	0,056014	0,028005
16	17	0,0817	0,0304	0,056	0,028	4,721095	0,056	0,028
14	18	0,035	0,013	0,2147	0,1067	4,703684	0,2147	0,1067
3	19	0,07	0,0261	0,056	0,028	4,7376	-0,49869	-0,27835
19	20	0,0475	0,034	0,056	0,028	4,733784	0,280106	0,14004
20	21	0,0817	0,0304	0,056	0,028	4,732637	0,056	0,028
20	22	0,0583	0,0217	0,168	0,084	4,731328	0,168	0,084
5	23	0,1017	0,0728	0,1133	0,0533	4,681391	0,1133	0,0533
5	24	0,0542	0,0388	0	0	4,670156	0,926763	0,453602

24	25	0,0933	0,0348	0,056	0,028	4,668829	0,056	0,028
24	26	0,0542	0,0388	0,1133	0,0533	4,656516	0,868412	0,423925
26	27	0,0949	0,068	0,056	0,028	4,635715	0,752016	0,368406
27	28	0,1085	0,0777	0,1867	0,0933	4,619056	0,524617	0,257727
28	29	0,0678	0,0486	0,168	0,0827	4,612366	0,337468	0,164106
29	30	0,0678	0,0486	0	0	4,609017	0,169356	0,081325
30	31	0,0678	0,0486	0,056	0,028	4,607898	0,056	0,028
27	32	0,1517	0,0565	0	0	4,629171	0,169411	0,081341
32	33	0,3733	0,1391	0,056	0,028	4,623807	0,056	0,028
32	34	0,0583	0,0217	0,1133	0,0533	4,627494	0,1133	0,0533
30	35	0,0583	0,0217	0,1133	0,0533	4,607333	0,1133	0,0533

Tabela VI: Dados dos Geradores Instalados no alimentador de trinta e seis barras

Barra	$P_G(MW)$	fp_{min}	fp_{max}	$V_{min}(kV)$	$\bar{V}(kV)$	$V_{max}(kV)$
7	0,625	0,85	0,95	4,08	4,741	5,52
19	0,835	0,82	0,9	4,08	4,7376	5,52

O algoritmo conseguiu chegar a um resultado ótimo de potência reativa que permitiu manter a tensão controlada no valor que se queria. O resultado é apresentado na tabela a seguir:

Tabela VII: Resultados dos Geradores

Barra	$Q_{min}(Mvar)$	$Q(Mvar)$	$Q_{max}(Mvar)$	$\bar{V}(kV)$	$V(kV)$
7	0,205428	0,274793	0,38734	4,741	4,741
19	0,404409	0,447209	0,582834	4,7376	4,7376

5 CONCLUSÃO

Neste trabalho, o MSP foi adaptado com sucesso para aplicação em sistemas de distribuição com geração distribuída.

O problema foi formulado para minimizar uma função erro, representada pela soma da diferença entre a tensão desejada e a tensão na barra elevada ao quadrado, sendo a tensão na barra dependente da potência reativa injetada no gerador, onde foi usado no algoritmo a função *fmincon* do *Matlab*[®] para implementar a restrição de injeção de reativos.

Para testar o algoritmo foram usados alimentadores disponíveis em outras fontes e em ambos os casos os valores de tensão convergiram para o valor esperado.

O código apesar de está rodando bem e funcionando perfeitamente, ainda pode ser melhorado, para que assim o tempo de execução do mesmo diminua, uma vez que atualmente está demorando cerca de um minuto para apresentar um resultado completo.

BIBLIOGRAFIA

SOUZA, B. A. ; BRAZ, H. D. M. ; ALBUQUERQUE, J. M. C. A. ; GUTTERRES, J. G. G. . **Fluxo de Carga em Sistemas de Distribuição Radiais com Geração Distribuída: Método da Soma de Potência Modificado**. Revista IEEE América Latina, v. 4, p. 48-56, 2006.

SOUZA, B. A. ; ALVES, H. N. ; ALMEIDA, A. M. F. . **Um algoritmo genético para localização ótima de reguladores de tensão em redes de distribuição**. In: XV Congresso Brasileiro de Automática, 2004, Gramado, 2004.

ALBUQUERQUE, J. M. C. . **Distribuição de Energia Elétrica: Fluxo de Carga em Sistemas de Distribuição**, 2005.

SOUZA, B. A. . **Como Funciona o MSP e o Que Precisa Ser Modificado?** UFCG, 2005

MONTICELLI, A. J. **Fluxo de carga em redes de energia elétrica**. São Paulo: Edgard Blücher, 1983.

STEVENSON, W. D. **Elementos de análise de sistemas de potência**. 2ª Edição. São Paulo: McGraw-Hill, 1986.

APÊNDICE A

```

clear all

dados = xlsread('Teste13barras.xls');

V0 = dados(1,1);
qtdbgeracao = dados(1,2);
Inicio = dados(2,:);
Trechos = dados(3,:);
R = dados(4,:);
X = dados(5,:);
P = dados(6,:);
Q = dados(7,:);
bgeracao = [0,dados(8,1:qtdbgeracao)];
PGeracao = [0,dados(9,1:qtdbgeracao)];
QGmin = [0,dados(10,1:qtdbgeracao)];
QGmax = [0,dados(11,1:qtdbgeracao)];
Vesp = dados(12,1:qtdbgeracao);
QG0 = (QGmin + QGmax)/2;
kgeracao = qtdbgeracao + 1;
OPTIONS = optimset('Algorithm','interior-point','TolFun',1e-9);

QGotimo = fmincon(@delV4,QG0,[],[],[],[],QGmin,QGmax,[],OPTIONS);

%Inicialização de Variáveis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
panterior = 0;
patual = 0;
qanterior = 0;
qatual = 0;
tol1 = 1;
tol2 = 1;
Fim = Trechos;
Tipo = zeros(length(Fim),1);
Nivel = zeros(length(Fim),1);
A = zeros(length(Trechos),1);
B = zeros(length(Trechos),1);
DeltaP = zeros(length(Trechos),1);
DeltaQ = zeros(length(Trechos),1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

k = 1;

% Determinação do Tipo de Cada Barra
while (k <= length(Inicio))
    flagtipo = 0;
    k1 = 1;
    while(k1 <= length(Fim))
        if(Fim(k) == Inicio(k1))
            flagtipo = flagtipo + 1;
        end
        k1 = k1 + 1;
    end

    if(flagtipo == 0)
        Tipo(k) = 0;
    elseif(flagtipo == 1)
        Tipo(k) = 1;
    else
        Tipo(k) = 2;
    end
end

```

```

    %Parte da Determinação do Nível
    if(Inicio(k) == 0)
        Nivel(k) = 1;
    end

    k = k + 1;
end

k = 1;

%Determinação do Nível de Cada Barra

while(k <= length(Inicio))

    if(Nivel(k) ~= 1)
        k1 = 1;
        while(k1 <= length(Inicio))
            if(Inicio(k) == Fim(k1))
                Nivel(k) = Nivel(k1) + 1;
            end
            k1 = k1 + 1;
        end
    end
    k = k + 1;
end

k2 = 1;
while(tol1 >= 1e-9 || tol2 >= 1e-9)

    %Início da primeira parte
    Tensoes = zeros(length(Trechos),1);
    PAux = zeros(length(Trechos),1);
    QAux = zeros(length(Trechos),1);
    NivelAux = max(Nivel);
    k = length(Trechos);
    k1 = length(Trechos);
    kgeracao = length(bgeracao);

    %Inicialização das Cargas Ativas e Reativas
    while (NivelAux > 1)
        while (k >= 1)

            if(Tipo(k) == 0)
                if(k == bgeracao(kgeracao))
                    PAux(k) = P(k) - PGeracao(kgeracao);
                    QAux(k) = Q(k) - QGotimo(kgeracao);
                    kgeracao = kgeracao - 1;
                else
                    PAux(k) = P(k);
                    QAux(k) = Q(k);
                end
            end

            if(Tipo(k) == 1)
                if(k == bgeracao(kgeracao))
                    PAux(k) = P(k) + PAux(k + 1) + DeltaP(k + 1) -
                    PGeracao(kgeracao);

```

```

        QAux(k) = Q(k) + QAux(k + 1) + DeltaQ(k + 1) -
QGotimo(kgeracao);
        kgeracao = kgeracao - 1;
    else
        PAux(k) = P(k) + PAux(k + 1) + DeltaP(k + 1);
        QAux(k) = Q(k) + QAux(k + 1) + DeltaQ(k + 1);
    end
end

if(Tipo(k) == 2)
    while (k1 >= 1)
        if(Inicio(k1) == Fim(k))
            PAux(k) = PAux(k) + PAux(k1) + DeltaP(k1);
            QAux(k) = QAux(k) + QAux(k1) + DeltaQ(k1);
        end
        k1 = k1 - 1;
    end
    if(k == bgeracao(kgeracao))
        PAux(k) = P(k) + PAux(k) - PGeracao(kgeracao);
        QAux(k) = Q(k) + QAux(k) - QGotimo(kgeracao);
        kgeracao = kgeracao - 1;
    else
        PAux(k) = P(k) + PAux(k);
        QAux(k) = Q(k) + QAux(k);
    end
    k1 = length(Trechos);
end
k = k - 1;
end
NivelAux = NivelAux - 1;
end

%Início do Trecho de Cálculo das Tensões e Perdas

while (NivelAux <= (max(Nivel) + 1))
    k = 1;
    while(k <= length(Trechos))
        if(Nivel(k) == NivelAux)
            if(Nivel(k) == 1)
                A(k) = (V0^2)/2 - (R(k)*PAux(k) + X(k)*QAux(k));
                B(k) = (R(k)^2 + X(k)^2)*(PAux(k)^2 + QAux(k)^2);
                Tensoes(k) = sqrt(A(k) + sqrt(A(k)^2 - B(k)));
                DeltaP(k) = (R(k)*(PAux(k)^2 + QAux(k)^2))/Tensoes(k)^2;
                DeltaQ(k) = X(k)*DeltaP(k)/R(k);
            else
                k1 = 1;
                while (k1 <= length(Trechos))
                    if(Inicio(k) == Fim(k1))
                        A(k) = (Tensoes(k1)^2)/2 - (R(k)*PAux(k) +
X(k)*QAux(k));
                        B(k) = (R(k)^2 + X(k)^2)*(PAux(k)^2 + QAux(k)^2);
                        Tensoes(k) = sqrt(A(k) + sqrt(A(k)^2 - B(k)));
                        DeltaP(k) = (R(k)*(PAux(k)^2 +
QAux(k)^2))/Tensoes(k)^2;
                        DeltaQ(k) = X(k)*DeltaP(k)/R(k);
                    end
                    k1 = k1 + 1;
                end
            end
        end
        k = k + 1;
    end
end

```

```
end
NivelAux = NivelAux + 1;
end

panterior = patual;
patual = sum(DeltaP);

qanterior = qatual;
qatual = sum(DeltaQ);

if(k2 >= 2)
    tol1 = ((patual - panterior)/patual)*100;
    tol2 = ((qatual - qanterior)/qatual)*100;
end

k2 = k2 + 1;

end

resultados = [DeltaP'; DeltaQ'; PAux'; QAux'; A'; B'; Tensoes'];
xlswrite('resultados13barras.xls', resultados, 'Resultados', 'B1');
```