Universidade Federal do Pará
Instituto de Tecnologia
Faculdade de Engenharia da Computação e Telecomunicações

3^a Avaliação - Polinômio de Lagrange de 3^a Ordem

Danilo Souza - 201006840008
Iago Medeiros - 201006840018

Belém, 15 de Dezembro de 2014

List of Tables

Chapter 1

Introdução

A palavra interpolar pode ser definida como o ato de introduzir partes de algo para completar o todo. Na matemática esse conceito é utilizado para encontrar funções que definem o comportamento de um dado fenômeno para todos os valores dentro de um dado intervalo a partir de amostras desse fenômeno, por exemplo, medir o poder calorifico de um gás para 30 valores de temperatura (e.g, [0,1,2,3,...,30]) e então extrair desses valores uma função f(x) capaz de calcular dentro da margem de erro desejada o poder calorífico da água para qualquer temperatura dentro do intervalo medido.

A função utilizada para interpolar os pontos conhecidos e então encontrar uma função confiável capaz de representar o problema é fundamental para que a interpolação seja possível na prática e essa função é a função polinomial, pois polinômios são a forma mais usada na matemática para calcular aproximações de funções por serem o único tipo de função que é possível encontrar usando apenas as 4 operações elementares (adição, multiplicação, subtração e divisão).

Existem diversos tipos de interpolação por polinômios (e.g, polinômio de ordem n, polinômio de Lagrange, fórmula de Newton, fórmula de Gregory-Newton, etc.), este trabalho irá mostrar nas seções seguintes a definição e as principais vantagens e desvantagens do Polinômios de Lagrange e apresentar um código em $MatLab^{©}$ que encontra o resultado da interpolação usando o polinômio de Lagrange para n pontos.

1.1 Histórico

O polinômio de Lagrange foi descoberto e proposto originalmente por Edward Waring (1736-1798), matemático inglês. Seu trabalho, *Problems Concerning Interpolation*, foi publicado em 1779 no journal *Philosophical Transactions of the Royal Society of London*[5], e abordava um método de diferenças de logaritmos interpoladores resolvido por meio de uma forma mais geral.

Anos depois, em 1783, o matemático e físico suíço, Leonhard Euler, chegou a conclusões bem próximas sobre o mesmo método. Os trabalhos de Euler foram amplamente estudados e expandidos pelo matemático e astrônomo italiano Joseph-Louis Lagrange. Lagrange, tendo como base os trabalhos de Euler, chegou as conclusões sobre a técnica de interpolação e publicou seu trabalho em 1795. A partir dessa data, o mundo passou a referenciar esta técnica como O Polinômio de Interpolação de Lagrange, mesmo que este não tenha sido o primeiro a descobri-lo. Outros trabalhos que culminaram da vertente Euler-Lagrange estão a equação de Euler-Lagrange e o Teorema de Fermat-Lagrange.

1.2 O polinômio de Lagrange

O polinômio de Lagrange é definido conforme equação 1.1:

$$
\begin{aligned}
P_0 &= (x - x_1)(x - x_2)\cdots(x - x_n) \\
P_1 &= (x - x_0)(x - x_2)\cdots(x - x_n) \\
P_2 &= (x - x_0)(x - x_1)\cdots(x - x_n) \\
&\vdots \\
P_n &= (x - x_0)(x - x_1)\cdots(x - x_{n-1})
\end{aligned}
\tag{1.1}
$$

As propriedades mostradas a seguir se aplicam ao polinômio de Lagrange:

$$
\begin{aligned}
P_i(x_j) = 0; i \neq j \\
P_i(x_j) \neq 0; i = j
\end{aligned}
\tag{1.2}
$$

Fazendo uma combinação linear dos polinômios da equação 1.1 obtém-se:

$$
P_n(x) = b_0p_0(x) + b_1p1(x) + \cdots + b_kp_k(x) + \cdots + b_np_n(x)
\tag{1.3}
$$

Onde $b_0, b_1, b_2, \cdots, b_n$ são constantes a serem determinadas

Generalizando a solução de b, temos:

$$
b_i = \frac{y_i}{P_i(x_i)}
\tag{1.4}
$$

Juntando as equações 1.3 e 1.4 temos que:

$$
\begin{aligned}
P_n(x) = \sum_{i=0}^{n} \frac{y_iP_i(x)}{P_i(x_i)} \\
P_n(x) = \sum_{i=0}^{n} y_i \prod_{j=0}^{n} {}_{i\neq j} \frac{(x - x_j)}{(x_i - x_j)}
\end{aligned}
\tag{1.5}
$$

1.3 Porque Utilizar Lagrange

Segundo [3] as principais vantanges e desvantagens de realizar interpolação usando o Método do Polinômio de Lagrange são:

- Vantagens

  - É tão eficiente quanto o Método de Newton quando realizada somente uma interpolação

- Desvantagens
  - Quando há necessidade de mais de uma interpolação o método começa a apresentar lentidão devido ao excesso de cálculos
  - Quando um termo é adicionado, é necessário recalcular os valores de $P_i(x)$

A tabela 1.1 mostra a complexidade das operações no Método do Polinômio de Lagrange e a tabela 1.2 mostra um comparativo entre este método e os métodos de Newton e Gregory-Newton.

Operações	**Complexidade**
Adições	$2n^2 + 3n + 1$
Multiplicações	$2n^2 + 3n + 1$
Divisões	$n + 1$

Table 1.1: Complexidade do Método de Interpolação usando Polinômio de Lagrange

Método	**Complexidade por operações**		
	Adições	Multiplicações	Divisões
Lagrange	$2n^2 + 3n + 1$	$2n^2 + 3n + 1$	$n + 1$
Gregory-Newton	$\frac{1}{2}n^2 + \frac{7}{2}n + 2$	n	$n + 1$
Newton	$n^2 + 3n$	n	$\frac{1}{2}n^2 + \frac{1}{2}n$

Table 1.2: Comparativo entre os Métodos de Lagrange, Gregory-Newton e Newton

Chapter 2

O código implementado

O código do $MatLab^{©}$ será mostrado abaixo. É importante notar que decidimos implementar um código de Polinômio de Lagrange que atende não só a equações de terceiro grau, mas também a qualquer polinômio de grau n, desde que sejam informados os valores de entrada corretos no código :

```

function [ p ] = lagrange( x_inicial ,y_inicial , valor)
%Lagrange - Funcao de interpolacao usando polinomio de Lagrange para n
%pontos
%   Funcao para calcular o resultado do polinomio de lagrange de grau n
%   para um valor especifico. Recebe dois vetores com os valores de x
%   e y dos pontos conhecidos e o valor x que se deseja obter f(x).
n = length(x_inicial);
dif(1,n) = 0;
produto = ones(1,n);
mat_dif(n,n) = 1;
x = valor;
prod_x = 1;
for i=1:n
    %Calcula a diferenca x - x(i); i=1,2,...,n (Usada para calcular o Prod_x)
    dif(1,i) = x - x_inicial(i);
    % Calcula o valor de Prod_x. O valor e atualizado a cada iteracao
    % levando em conta o proximo valor de x(i).
    prod_x = prod_x*dif(i);
    for j=1:n
        if i==j
            mat_dif(i,j) = 1;
        else
            % Constroi a matriz com as diferencas entre x(i) - x(j)
            mat_dif(i,j) = x_inicial(i)-x_inicial(j);
        end
        % Usa o resultado da matriz de diferencas para calcular Prod(i)
        produto(1,i) = produto(1,i)*mat_dif(i,j);
    end
end

%mat_dif(find(eye(3))) = [1 1 1];
resultado = zeros(1,n);

% Calcula cada termo da solucao
for i=1:n
    resultado(1,i) = y_inicial(1,i)/(dif(1,i)*produto(1,i));
end

% Formula final
p = prod_x*(sum(resultado));

end
```

Listing 2.1: Código Implementado

Chapter 3

Aplicações na Indústria

O polinômio de Lagrange é usado para fazer cálculos rápidos [1] de polinômios interpoladores na mão, devido a sua velocidade e bom desempenho. Ele também foi testado em um ADC (conversor analógico-digital) na área de processamento digital de sinais [4], quando precisaram reconstruir o sinal digital a partir de amostras de sinal analógico em tempo real. Alguns trabalhos na área de saúde envolvem utilizar o método de Lagrange para analisar um padrão de fadiga muscular no ser humano [2].

Chapter 4

Conclusão

Conforme mostrado neste artigo o Método de Lagrange para interpolação de funções não possui eficiência ótima para situações em que se necessita adicionar termos em tempo real, entretanto seu desempenho é satisfatório para aplicações de tempo real, por exemplo, como visto em [4], dado que, uma vez que o sinal é recebido, só será interpolado uma vez, não havendo assim a necessidade de adição de termos, o que torna o Polinômio de Lagrange uma boa alternativa.

Além de sua eficiência computacional, o polinômio de Lagrange é bem simples de ser implementado conforme pode-se verificar no código apresentado neste trabalho, o que torna sua utilização ainda mais interessante em sistemas embarcados que possuem pouca capacidade de processamento e armazenamento.

Bibliography

[1] Marina Andretta. Interpolação polinomial: Polinômio de lagrange. `http://www.icmc.usp.br/pessoas/andretta/ensino/aulas/sme0500-1-12/iplagrange.pdf`, mai 2012.

[2] David Gabriel, David Proctor, Dean Engle, Sreekumaran Nair, Janet Vittone, and Kai-Nan An. Application of the lagrange polynomial in skeletal muscle fatigue analysis. *Research Quarterly for Exercise and Sport*, 73(2):168–174, 2002. PMID: 12092891.

[3] Flaviane Venditti. Método de interpolação de lagrange. `http://www2.dem.inpe.br/mcr/Inpe/CMC-203-0/pdf/Flaviane.pdf`, dez 2014.

[4] Vladan Vučković. The reconstruction of the compressed digital signal using lagrange polynomial interpolation. *16th Telecommunications forum*, nov 2008. Acessado em 04/12/2014.

[5] Edward Waring. Problems concerning interpolations. *Philosophical Transactions of the Royal Society of London*, jan 1779. Accessado em 09/12/2014.