

MANUAL DO PROGRAMA NEWTOP 1.0

Álvaro Ferreira Marques Azevedo¹

Abril de 2000

1 - Introdução

O conjunto de programas, cuja designação genérica é NEWTOP, destina-se à resolução de programas matemáticos não lineares pelo método de Lagrange-Newton. As derivadas de primeira e segunda ordem são calculadas analiticamente pelo próprio programa de um modo eficiente e exacto. Apresentam-se em seguida as principais capacidades e limitações deste *package*.

A função objectivo, as restrições desigualdade e as restrições igualdade têm de ser definidas por intermédio de polinómios generalizados. Para exemplificar a estrutura de um destes polinómios apresenta-se o seguinte exemplo:

$$f(x_1, x_2, x_3) = -4.9 * x_3^{-2} + 0.5 * x_1^3 * x_2 * x_3^{-1} - 6.8 + 2.9 * x_2$$

Neste tipo de expressões não são aceites parênteses, divisões ou expoentes não inteiros. Sempre que uma expressão não se enquadre no tipo de polinómios aceites pelo programa é necessário substituir as expressões por outras equivalentes, eventualmente com recurso a novas variáveis auxiliares. Apresenta-se em seguida um exemplo:

$$\frac{x^3 - 5.7}{9.8 * y^2 - z} - 3.1 < 0 \quad \Leftrightarrow \quad \begin{cases} -D + 9.8 * y^2 - z = 0 \\ x^3 * D^{-1} - 5.7 * D^{-1} - 3.1 < 0 \end{cases}$$

As restrições igualdade têm de terminar com: = 0 ;

As restrições desigualdade têm de terminar com: < 0 ;

2 - Módulos que constituem o *package* NEWTOP

Para efectuar a resolução de um problema de optimização com o *package* NEWTOP é necessário executar os seguintes módulos, cuja descrição será efectuada mais adiante:

- nlpconv
- preopt
- newtop
- posopt
- viewopt (opcional)

¹Professor Auxiliar da Faculdade de Engenharia da Universidade do Porto - Portugal
URL: <http://www.fe.up.pt/~alvaro>

3 - Ficheiros de dados com a descrição do programa matemático

jobname_np.dat (*non linear program*): ficheiro cujo conteúdo descreve a função objectivo, as restrições desigualdade e as restrições igualdade. O jobname é escolhido pelo utilizador, não devendo ser utilizados caracteres acentuados ou espaços. O ficheiro jobname_np.dat é lido pelo módulo nlpcnv e é constituído pelos seguintes componentes:

```
Título
Min.
  Função objectivo
s.t.i.c.
  Restrições desigualdade
s.t.e.c.
  Restrições igualdade
END_OF_FILE
```

Todos os caracteres situados à direita do caracter '#' são ignorados.

Podem não existir restrições (desigualdade, igualdade ou ambas).

Cada restrição é precedida de um título curto, situado na linha anterior, com um máximo de 25 caracteres e terminado com o caracter ': '.

A função objectivo e as restrições têm de ser terminadas com o caracter '; '.

Todos os componentes das funções têm de estar situados entre delimitadores (espaços, TAB's ou caracteres de fim de linha).

Exemplo de um ficheiro do tipo jobname_np.dat:

```
### Main title of the nonlinear program
Optimization of a cantilever beam with 2 load cases

#           Max. stress = 250           | P1
#   /|                                     |
#   /| A                                 |
#   /+-----+                           \ \ / /
#   /|                                     | B
#   /|                                     |-----+
#   /| C                                 | D
#   /|                                     | O P2
#   /|                                     |
#   /+-----+
#   /|
#   /|
#           L1           L2
#   |-----|-----|
#
#           L
#   |-----|

Min.
  L1 * B1 * H1 + L2 * B2 * H2 ;

s.t.i.c.

Stress (A):
  6 * P1 * L * B1 ^ -1 * H1 ^ -2 - 250 < 0 ;

Stress (B):
```

```

        6 * P1 * L2 * B2 ^ -1 * H2 ^ -2 - 250 < 0 ;

Stress (C):
        6 * P2 * L * B1 ^ -2 * H1 ^ -1 - 250 < 0 ;

Stress (D):
        6 * P2 * L2 * B2 ^ -2 * H2 ^ -1 - 250 < 0 ;

# Side constraints

L1 min.:
    - L1 + 0.1 < 0 ;

L2 min.:
    - L2 + 0.1 < 0 ;

s.t.e.c.

Total length:
    L - L1 - L2 = 0 ;

END_OF_FILE

```

`jobname_vn.dat` (*variable names*): ficheiro em que são declarados os nomes das variáveis. O `jobname` tem de coincidir com o que é utilizado nos restantes ficheiros que descrevem o problema de optimização. Se o ficheiro `jobname_vn.dat` não existir, o programa assume que as variáveis são as seguintes: x_1, x_2, \dots, x_n . Quando o ficheiro `jobname_vn.dat` existe, a sua leitura é efectuada pelo módulo `nlpconv`. Cada variável pode ter um nome constituído por um máximo de 25 caracteres consecutivos (e.g., `force_79_3`).

Exemplo de um ficheiro do tipo `jobname_vn.dat`:

```

### Main title of the variable names
Cantilever optimization - variable names

### Parameters of the variable names
9 # nvarn (number of variable names)

### Variable name
1   P1
2   P2
3   L1
4   L2
5   L
6   B1
7   H1
8   B2
9   H2

END_OF_FILE

```

`jobname_su.dat` (*substitutions*): ficheiro em que podem ser especificadas substituições de variáveis. O `jobname` tem de coincidir com o que é utilizado nos restantes ficheiros que descrevem o problema de optimização. O ficheiro `jobname_su.dat` é opcional. Quando existe é lido pelo módulo `preopt`.

Apresentam-se em seguida dois exemplos de substituições de variáveis:

$$Area = 12.9$$

$$B_{15} = 0.45 * H_{15}$$

Exemplo de um ficheiro do tipo `jobname_su.dat`:

```
### Main title of the variable substitution
Cantilever optimization - variable substitution

### Substitution parameters
4 # nvsub (number of variables to be substituted)

### Substitutions
1 B1 = 1.5 * B2 # B1 is linked to B2
2 L = 7.0
3 P1 = 10.0
4 P2 = 4.0

END_OF_FILE
```

`jobname_is.dat` (*initial solution*): ficheiro em que são especificados os valores iniciais das variáveis, bem como o tipo de cada variável. O `jobname` tem de coincidir com o que é utilizado nos restantes ficheiros que descrevem o problema de optimização. O ficheiro `jobname_is.dat` tem de existir, sendo lido pelo módulo `preopt`. Os possíveis tipos de variáveis são:

- i - Variáveis independentes: o seu valor inicial deve ser cuidadosamente pré-dimensionado com o objectivo de aumentar a eficiência do processo iterativo de resolução do programa não linear. A selecção de valores com ordens de grandeza muito distintas das que são esperadas na solução óptima pode comprometer a convergência do processo iterativo.
- d - Variáveis dependentes: em certos tipos de problemas o valor inicial das variáveis dependentes pode ser calculado com o próprio programa NEWTOP. Os valores iniciais que são especificados neste ficheiro devem ser cuidadosamente pré-dimensionados, excepto no caso em que, após a aplicação das substituições especificadas no ficheiro `jobname_su.dat` e após a substituição das variáveis independentes pelo seu valor inicial, o conjunto das restrições igualdade constitua um sistema de equações lineares. Neste caso, o valor colocado no ficheiro `jobname_is.dat` pode ser nulo, sendo necessário especificar no ficheiro `jobname_oo.dat` que se pretende que o módulo `newtop` calcule o valor inicial das variáveis dependentes. Em qualquer caso, o número de variáveis dependentes tem de coincidir com o número de restrições igualdade.
- s - Variáveis substituídas: o seu valor inicial é irrelevante, podendo ser sempre nulo. Estas variáveis são removidas do programa matemático, sendo substituídas pelo valor/expressão especificado no ficheiro `jobname_su.dat`.

Exemplo de um ficheiro do tipo `jobname_is.dat`:

```
### Main title of the initial solution
Cantilever optimization - initial solution

### Initial solution and design variable flag (i/d/s)
### (i) independent ; (d) dependent ; (s) substituted
1 0.0 s # P1
2 0.0 s # P2
3 3.5 i # L1
```

```

4      0.0      d      #      L2
5      0.0              s      #      L
6      0.0              s      #      B1
7      1.5      i              #      H1
8      0.6      i              #      B2
9      1.2      i              #      H2

```

END_OF_FILE

jobname_oo.dat (*optimization options*): ficheiro em que podem ser especificados alguns parâmetros e opções relativos ao processo iterativo. O **jobname** tem de coincidir com o que é utilizado nos restantes ficheiros que descrevem o problema de optimização. O ficheiro **jobname_oo.dat** é opcional. Quando existe é lido pelo módulo **newtop**. Quando não existe são admitidas as opções por defeito, que são as que se encontram no ficheiro exemplo abaixo indicado.

Exemplo de um ficheiro do tipo **jobname_oo.dat**:

```

### Main title of the optimization options
Title of the optimization options

### Optimization options
#-----
# General options:
y      # scaxc (y)      scaling with x coefficients (y/n)
y      # scasc (y)      scaling with slack coefficients (y/n)
y      # scahc (y)      scaling with homogenizing coefficients (y/n)
1.0    # vlam1 (1.0)    minimum value of the initial Lagrange multipliers
1.0    # vlam2 (1.0)    maximum value of the initial Lagrange multipliers
1.0e-5 # tolre (1.0e-5) tolerated residual in the gradient of Lagrangean
20     # miter (20)     maximum number of iterations
1.0e-2 # tolrb (1.0e-2) tolerated residual before the last iteration
1.0e-4 # tolxm (1.0e-4) tolerated x minimum value
1.0e-6 # tolpi (1.0e-6) tolerated minimum pivot
1.0e-12 # tolze (1.0e-12) smallest number tolerated as zero
n      # chsle (n)      check solution of system of linear equations (y/n)
3      # nbime (3)      number of iterations of the bisection method
1.0e-3 # asmin (1.0e-3) alpha step minimum value
1.0    # asmax (1.0)   alpha step maximum value
1.0e-6 # delnu (1.0e-6) delta used to calculate numerical derivatives
#-----
# Dependent design variables options:
y      # cdedv (y)      calc. initial value of dependent design var.s (y/n)
1.0e-5 # tolrd (1.0e-5) tolerated residual in the equality constraints
20     # mited (20)     maximum number of iterations
1.0e-12 # tolzd (1.0e-12) smallest number tolerated as zero
n      # chsld (n)      check solution of system of linear equations (y/n)
3      # nbimd (3)      number of iterations of the bisection method
1.0e-3 # asmid (1.0e-3) alpha step minimum value
1.0    # asmad (1.0)   alpha step maximum value
1.0e-6 # delnd (1.0e-6) delta used to calculate numerical derivatives
#-----
# Conjugate gradients options:
n      # conjg (n)      try conjugate gradients to solve H dx + g = 0 (y/n)
1.0e-6 # redec (1.0e-6) preconditioned residual decay
10.0   # fmite (10.0)  factor that multiplies n.eq. to obtain max.n.iter.
#-----
END_OF_FILE

```

4 - Passos para a resolução de um problema de optimização

4.1 - Tarefas realizadas por cada módulo

Tal como foi referido na Secção 2, são os seguintes os passos que é necessário executar para resolver um problema de optimização com o *package* NEWTOP:

```
nlpconv  
preopt  
newtop  
posopt  
viewopt (opcional)
```

O módulo `nlpconv` destina-se apenas a interpretar a descrição do programa matemático.

Quando não se pretendem efectuar substituições, o módulo `preopt` apenas processa a solução inicial. Quando se pretendem efectuar substituições, o respectivo processamento é também efectuado pelo módulo `preopt`.

O módulo `newtop` efectua todas as operações correspondentes ao processo iterativo pelo método de Lagrange-Newton. Opcionalmente, é possível visualizar um gráfico com a evolução de alguns parâmetros e é possível introduzir uma pausa no fim de cada iteração para se observar o estado do processo iterativo.

Os resultados finais são convertidos para um formato visível pelo módulo `posopt`. Pelo respectivo menu é ainda possível seleccionar algumas operações auxiliares.

O módulo `viewopt` é opcional e destina-se apenas a representar graficamente a evolução de algumas grandezas ao longo do processo iterativo. As grandezas que são representadas são as seguintes:

Line search parameter - parâmetro de pesquisa unidimensional. Está associado à "rapidez" da convergência. Deve assumir valores tão elevados quanto possível para que o número de iterações seja baixo e a convergência seja rápida. O seu valor máximo é controlado pelo parâmetro `asmax`, que nunca deve ser superior à unidade. Se ao parâmetro `asmax` for atribuído um valor maior do que o ideal, pode não ser possível obter uma convergência do processo iterativo.

Equivalent residual - resíduo equivalente. Destina-se a avaliar a grandeza do erro associado à iteração corrente. Se o processo iterativo decorrer sem sobressaltos, este indicador deve decrescer monotonamente até um valor inferior ao do parâmetro `tolre` (ver o ficheiro `jobname_oo.dat`).

Objective function - função objectivo. Em problemas de minimização deve decrescer até ao valor óptimo.

Histograma - variação de todas as variáveis envolvidas no processo iterativo (variáveis de projecto, variáveis de desvio, multiplicadores de Lagrange das restrições desigualdade e multiplicadores de Lagrange das restrições igualdade). Um valor unitário significa que a variável manteve o seu valor inicial. Um valor negativo significa que a variável mudou de sinal. Com este histograma torna-se simples observar graficamente

se o multiplicador de Lagrange de alguma restrição desigualdade se tornou negativo. (Nota: quando na solução final existe um ou mais valores negativos em multiplicadores de Lagrange associados a restrições desigualdade, a solução é admissível, mas existem garantidamente outras soluções admissíveis em que o valor da função objectivo é ainda menor.)

4.2 - Exemplo sem substituição de variáveis

Supondo que o jobname é viga, devem ser executados os seguintes passos:

```
nlpconv viga
preopt viga
newtop viga ou newtop -g viga ou newtop -gp viga
posopt viga
viewopt viga
```

O módulo nlpconv lê o ficheiro viga_np.dat. Lê também o ficheiro viga_vn.dat, caso ele exista e esteja situado no mesmo directório.

O módulo preopt lê o ficheiro viga_is.dat.

O módulo newtop lê o ficheiro viga_oo.dat, caso ele exista e esteja situado no mesmo directório. Se este ficheiro não existir são admitidos valores por defeito para todos os parâmetros. A opção 'g' corresponde à selecção de saídas gráficas. A opção 'p' corresponde à selecção de uma pausa no fim de cada iteração.

No módulo posopt a opção mais importante é a que gera o ficheiro viga_or.lpt contendo todos os resultados.

4.3 - Exemplo com substituição de variáveis

Supondo que o jobname é viga, devem ser executados os seguintes passos:

```
nlpconv viga
preopt viga viga_s
newtop viga_s ou newtop -g viga_s ou newtop -gp viga_s
posopt viga_s
viewopt viga_s
```

O módulo nlpconv lê o ficheiro viga_np.dat. Lê também o ficheiro viga_vn.dat, caso ele exista e esteja situado no mesmo directório.

O módulo preopt lê os ficheiros viga_is.dat e viga_su.dat. Após a substituição de variáveis o jobname passa a ser viga_s (o acréscimo da letra 's' indica que se trata do programa matemático já com as substituições efectuadas). Esta alteração do jobname permite que toda a informação correspondente ao programa matemático original seja preservada.

O módulo newtop lê o ficheiro viga_oo.dat, caso ele exista e esteja situado no mesmo directório. Se este ficheiro não existir são admitidos valores por defeito para todos os parâmetros. A opção 'g' corresponde à selecção de saídas gráficas. A opção 'p' corresponde à selecção de uma pausa no fim de cada iteração.

No módulo `posopt` a opção mais importante é a que gera o ficheiro `vigas_or.lpt` contendo todos os resultados.

4.4 - Notas

Os ficheiros com extensão `.dat` podem ser modificados com um qualquer processador de texto (e.g., `notepad`, `wordpad`, `vi`, `emacs`). Sempre que existam mais do que uma ou duas dezenas de funções, os ficheiros `.dat` devem ser gerados com *software* específico para cada tipo de problema de optimização (e.g., treliças, muros de suporte).

Os ficheiros com extensão `.aux` são informativos e podem ser visionados com um qualquer processador de texto.

Os ficheiros com extensão `.lpt` constituem os resultados e podem ser visionados com um qualquer processador de texto.

Os ficheiros com extensão `.g2d` destinam-se à produção de gráficos do tipo `x-y`, podendo ser visionados com um qualquer processador de texto.

Os ficheiros com extensão `.bin` e `.tmp` são do tipo binário e não podem ser visionados com um processador de texto.

Durante a execução dos diversos módulos, sempre que seja efectuada uma pergunta que tenha uma resposta por defeito, esta deverá ser aceite, a não ser que o utilizador esteja ciente das alterações que vai voluntariamente provocar.

Nos monómio das funções que são especificadas no ficheiro `jobname_np.dat`, a ordem pela qual as variáveis são multiplicadas umas pelas outras tem de coincidir com a ordem da sua declaração no ficheiro `jobname_vn.dat`. Por exemplo, se a declaração da variável `Area` preceder a declaração da variável `Comprimento`, o seu produto tem de figurar no ficheiro `jobname_np.dat` como `Area * Comprimento` e não `Comprimento * Area`.

Se não ocorrer convergência, as soluções mais aconselháveis são:

- 1 - No ficheiro `jobname_oo.dat`, baixar o valor do parâmetro `asmax` para, por exemplo, um valor de 0.1. Se esta alteração não resultar devem ser ensaiados outros valores de `asmax` no intervalo [0.01 , 1.0].
- 2 - Modificar a solução inicial no ficheiro `jobname_is.dat`. Para estimar quais são as alterações que devem ser efectuadas, aconselha-se uma consulta às soluções intermédias do processo iterativo problemático para tentar compreender o que é que correu mal. As soluções intermédias podem ser obtidas com o módulo `posopt` (opção 1), especificando o número da iteração pretendida. Esta consulta às soluções intermédias pode ser efectuada durante a execução do módulo `newtop`, bastando para isso abrir uma nova janela e arrancar aí o módulo `posopt`.

Agradecimentos

A preparação deste manual foi iniciada pelos alunos do 5ºano da Licenciatura em Engenharia Civil da FEUP Jorge Agostinho Pinho e Marco António Duarte. Agradeço a ambos o entusiasmo com que abordaram este tema tão "impopular".

Bibliografia

Azevedo, Álvaro F. M. - *Optimização de Estruturas com Comportamento Linear e Não Linear*, Dissertação para Doutoramento em Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, 1994.

<http://www.alvaroazevedo.com/publications/books/PhD/OE.pdf>

Azevedo, Álvaro F. M. - *Apontamentos da disciplina "Optimização de Estruturas"* - Mestrado em Estruturas de Engenharia Civil da Faculdade de Engenharia da Universidade do Porto, 1995.

<http://www.alvaroazevedo.com/publications/reports/1995/OE.pdf>

Fletcher, Roger - *Practical Methods of Optimization* (Second Edition) - John Wiley & Sons, 1987.

URL's

Manual do Programa NEWTOP 1.0 (este documento):

http://civil.fe.up.pt/Software/NEWTOP_V1_0/NEWTOP_1.0_Manual.pdf

Versão de demonstração do programa NEWTOP 1.0:

http://civil.fe.up.pt/Software/Downloads/NEWTOP/V1_0/Demo/NEWTOP_1.0.zip