

Uso di Make_em_file per EasyMesh

a cura di
Marco Politi

Milano, luglio 2006

E' stata messa a punto una function Matlab (Make_em_file.m) per semplificare la generazione del file .d nel formato richiesto dal programma EasyMesh. La function risulta utile in particolare per la reticolazione di geometrie comprendenti contorni curvilinei, che vanno approssimati tramite una spezzata.

Nel seguito ne viene descritto l'uso facendo riferimento alla geometria 2D mostrata in Fig. 1. Questa si riferisce a una linea di trasmissione di tipo coassiale, con un conduttore interno di raggio r_i e un conduttore esterno rettangolare, di dimensioni $2a \times 2b$ e spigoli arrotondati (raggio di curvatura r_e). Vengono considerati i casi di reticolazione sia della struttura completa che —date le simmetrie del problema— di un mezzo e un quarto della struttura stessa.

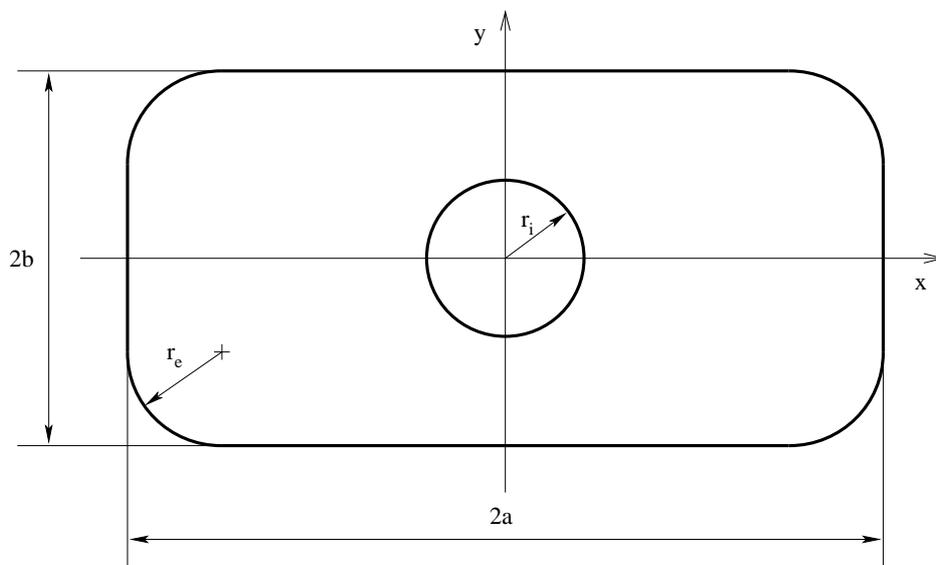


Figura 1: Sezione trasversale di una linea ideale di trasmissione.

Reticolazione dell'intera struttura

La gestione parametrica della reticolazione dell'intera struttura considerata può ottenersi a partire dall'individuazione dei nodi, estremi dei segmenti e degli archi di cerchio che ne costituiscono il contorno, come mostrato in Fig. 2. Si noti che nell'attuale versione di 'Make_em_file' le aperture degli archi di cerchio sono limitate a 90° , con orientamento legato agli assi cartesiani.

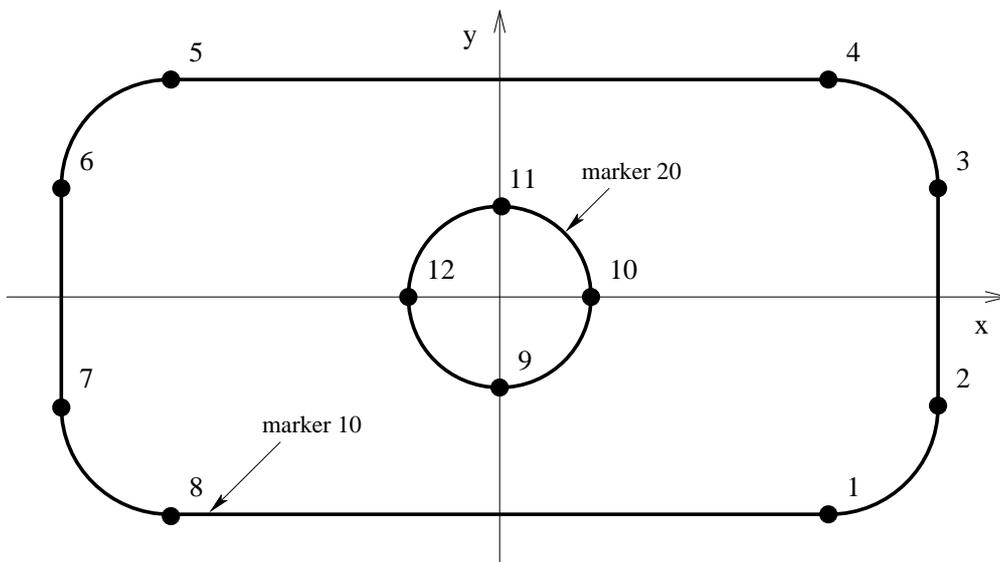


Figura 2: Nodi e marker per la descrizione dell'intera struttura.

Nel caso di N nodi, questi vengono numerati da 1 a N . La conversione alla numerazione richiesta da 'EasyMesh' (da 0 a $N - 1$) viene gestita in modo automatico dalla function 'Make_em_file'.

La reticolazione può allora ottenersi mediante lo stralcio di procedura Matlab riportato come Esempio 1 in Appendice. In questo, dopo aver definito le dimensioni geometriche della struttura (vedi Fig. 1), vengono fissati i parametri della reticolazione (dimensione dei lati 'dl', marker 'mk1' e 'mk2' da assegnare ai conduttori esterno e interno rispettivamente).

La function 'Make_em_file' viene quindi chiamata passandole una matrice 'dati' e il nome del file .d da generare (nell'esempio 'Es_ret1.d'). La matrice 'dati' comprende una successione di righe, una per ogni nodo della porzione di contorno che si vuole descrivere, dal primo all'ultimo nodo della porzione stessa.

Ogni riga comprende nell'ordine:

- 1) il numero del nodo;
- 2) la sua ascissa;
- 3) la sua ordinata;
- 4) la dimensione desiderata per i lati del reticolo in prossimità del nodo;
- 5) il marker del nodo;
- 6) il marker del lato tra il nodo considerato e il precedente ('-1' se il nodo è il primo di una sequenza);
- 7) l'angolo di apertura dell'arco tra il nodo considerato e il precedente ($+90^\circ$ per arco orientato in senso antiorario, -90° per arco orario, 0 per segmento);
- 8) il numero di segmenti della poligonale corrispondente (0 per segmento).

Si noti al riguardo che:

- il primo nodo di una porzione di contorno viene individuato tramite un '-1' in colonna 6;

- l'ultimo nodo di questa viene invece contassegnato dal suo indice cambiato di segno;
- possono essere descritte più porzioni di contorno con le convenzioni riportate sopra (nell'esempio 1-2-...-8-1 e 9-12-11-10-9);
- il verso orario con cui viene descritto il conduttore interno serve a escludere dalla reticolazione il conduttore stesso (vedi note relative all'uso di EasyMesh).

Reticolazione di mezza struttura

Come Esempio 2 è riportato in appendice lo stralcio di procedura Matlab per gestire la reticolazione di metà struttura. Si fa riferimento in questo ai nodi e ai marker di Fig. 2.

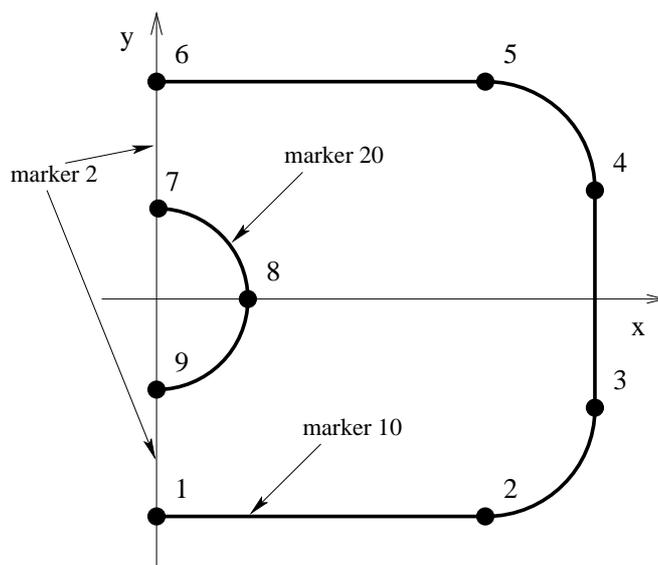


Figura 3: Come Fig. 2 per mezza struttura.

A differenza del caso precedente, basta un unico contorno chiuso per descrivere la geometria. Viene utilizzato un ulteriore marker ('2' nell'esempio) per identificare i piani di simmetria. La reticolazione che se ne ottiene è uniforme come in Esempio 1.

Reticolazione di un quarto di struttura

In Esempio 3 viene infine riportata la gestione parametrica della reticolazione di un quarto della struttura, con riferimento alla Fig. 4.

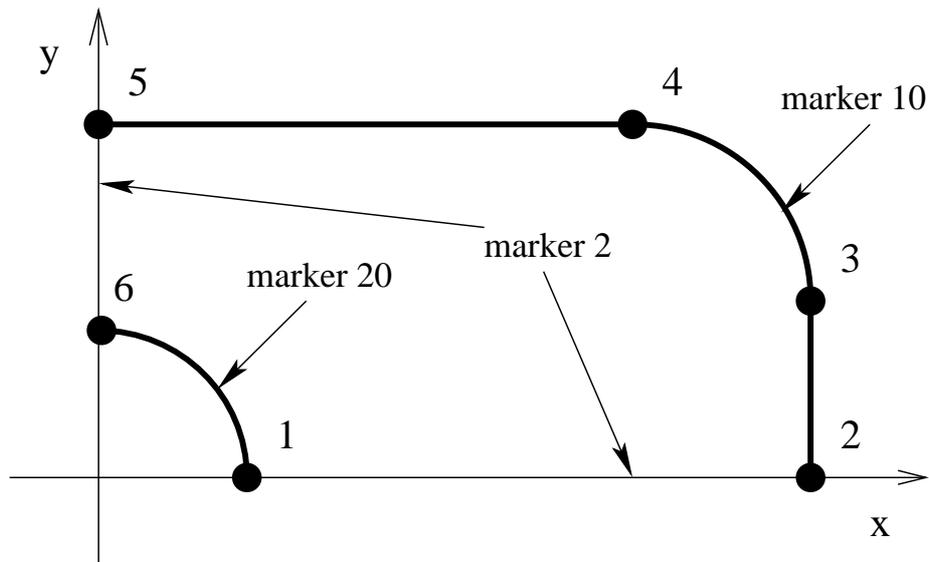


Figura 4: Come Fig. 2 per un quarto di struttura.

Valgono le considerazioni fatte nei casi precedenti. La sola differenza consiste nell'uso del parametro 'dl' per ottenere una reticolazione più fitta in corrispondenza del conduttore interno.

Appendice

Esempio 1

```
%
% Dimensioni geometriche (unità arbitrarie)
%

a = 12;
b = 6;
ri = 2.5;
re = 3;

%
% Parametri di reticolazione
%

dl = 1.5;
mk1 = 10;
mk2 = 20;

%
% Reticolazione 1 (struttura intera)
%

%      ni      xi      yi      dli      mkn      mkl      ang      nl
dati = [1      a-re  -b      dl      mk1      -1      0      0; ...
```

```

2    a    -b+re dl    mk1    mk1    90    3; ...
3    a    b-re  dl    mk1    mk1    0     0; ...
4    a-re b     dl    mk1    mk1    90    3; ...
5    -a+re b    dl    mk1    mk1    0     0; ...
6    -a    b-re  dl    mk1    mk1    90    3; ...
7    -a    -b+re dl    mk1    mk1    0     0; ...
8    -a+re -b    dl    mk1    mk1    90    3; ...
-1   a-re  -b    dl    mk1    mk1    0     0; ...

9    0    -ri   dl    mk2    -1     0     0; ...
12   -ri   0     dl    mk2    mk2    -90   4; ...
11   0     ri    dl    mk2    mk2    -90   4; ...
10   ri    0     dl    mk2    mk2    -90   4; ...
-9   0     -ri   dl    mk2    mk2    -90   4];

```

```

fname = 'Es_ret1';
Make_em_file(fname, dati);
eval(['!easymesh ' fname])

```

Esempio 2

```

%
% Parametri di reticolazione
%

dl = 2;
mk1 = 10;
mk2 = 20;

% Reticolazione 2 (mezza struttura)
%      ni   xi   yi   dli  mkn  mk1  ang  nl
dati = [1    0   -b   dl   mk1  -1   0   0; ...
        2   a-re -b   dl   mk1  mk1  0   0; ...
        3    a   -b+re dl   mk1  mk1  90  2; ...
        4    a   b-re  dl   mk1  mk1  0   0; ...
        5   a-re b    dl   mk1  mk1  90  2; ...
        6    0   b    dl   mk1  mk1  0   0; ...
        7    0   ri   dl   mk2   2   0   0; ...
        8   ri   0    dl   mk2  mk2  -90  4; ...
        9    0   -ri  dl   mk2  mk2  -90  4; ...
       -1   0   -b   dl   mk1   2   0   0];

fname = 'Es_ret2';
Make_em_file(fname, dati);
eval(['!easymesh ' fname])

```

Esempio 3

```
%
% Parametri di reticolazione
%

dl = 1.5;
mk1 = 10;
mk2 = 20;

% Reticolazione 3 (quarto di struttura)
%      ni    xi    yi    dli  mkn  mk1  ang  nl
dati = [1    ri    0    dl/2  mk2  -1    0    0; ...
        2    a     0    dl     mk1   2     0    0; ...
        3    a    b-re  dl     mk1  mk1    0    0; ...
        4    a-re  b     dl     mk1  mk1    90   4; ...
        5    0     b     dl     mk1  mk1    0    0; ...
        6    0     ri    dl/2  mk2   2     0    0; ...
        -1   ri    0     dl/2  mk1  mk2   -90   6];

fname = 'Es_ret3';
Make_em_file(fname, dati);
eval(['!easymesh ' fname])
```