# **GMSH: Generador de Mallas**

http://geuz.org/gmsh/

Mini Tutorial v1.1 - 19/Ene/2011 Mariano Re - mre@fi.uba.ar

### INTRODUCCIÓN

GMSH es un software para generar mallados bidimensionales y tridimensionales automáticamente<sup>1</sup>. GMSH fue concebido como un software académico para resolver problemas académicos, sin embargo a lo largo de los años se ha mejorado lo suficiente para ser utilizado fuera de entornos universitarios. GMSH es software libre, está escrito en C++ y es multiplataforma (Unix/Linux, Mac OS X, Windows).

### FUNCIONALIDADES DE GMSH

GMSH presenta cuatro módulos: geométrico, mallador, solver y postprocesado. El módulo geométrico (de diseño), permite dibujar visualmente una figura aunque con capacidades limitadas, a pesar de poder definirlas paramétricamente. El módulo de mallado se apoya en las definiciones geométricas (incluidas la longitudes características) para crear mallados en una, dos y tres dimensiones, con elementos de formas varias (líneas, triángulos, tetraedros, prismas, hexaedros y pirámides) y eligiendo el tipo de algoritmo de mallado. El módulo solver (de resolución) implica la llamada a ciertos sistemas externos con los que interactuar (al momento programadas por el usuario). El módulo de postprocesado (ligado a la visualización) incluye gran número de operaciones diferenciales, vectoriales, herramientas para trabajar con números complejos y funcionalidades para extraer datos de las distribuciones de datos de entrada.

### MODULO GEOMÉTRICO

El módulo geométrico permite dibujar puntos, líneas, superficies y volúmenes. Además permite seleccionar qué elementos están físicamente presentes en las figuras, puesto que en ocasiones es necesario definir elementos auxiliares sin sentido físico.

A partir de la definición de puntos en el espacio es posible ir construyendo líneas, círculos y elipses, con los que se pueden ir definiendo las superficies y/o volúmenes que constituyen la figura buscada. Una vez conseguida es muy recomendable definir los puntos, líneas y superficies que tienen sentido físico con *Physical Groups*, indicando al mallador que partes discretizar.

Cada uno de los elementos del dibujo se denominan *Entidades Geométricas,* y son los siguientes: Punto (*Point*) - Línea (*Line*) - Arcos de círculo (*Circle*) - Arcos de elipse (*Ellipse*) - Spline (*Spline*) -BSpline (*BSpline*) - Superficie plana (*Plane Surface*) - Superficies regladas (*Ruled Surface*) - Volúmenes (*Volumen*). Las herramientas de dibujo disponibles son: Translación (*Translate*) - Rotación (*Rotate*) -

<sup>&</sup>lt;sup>1</sup> Geuzaine, C., Remacle, J.F., 2009, Gmsh: a three-dimensional finite element mesh generator with built-in preand post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79 (11), 1309-1331.

Desdoblaje (Split) - Escalado (Scale) - Simetría (Symmetry) - Extrusión (Extrude) - Coherencia (Coherence).

La geometría del dominio a mallar puede ser definida desde un archivo de texto (formato propio *\*.geo*). Utilizando un lenguaje propio en ASCII se pueden crear figuras parametrizadas o incluir bucles. Por ejemplo las longitudes características de la densidad de mallado es un caso común de variable parametrizada.

#### MODULO DE MALLADO

El módulo de mallado agrupa varios algoritmos de mallado 1D, 2D y 3D, creando mallas pensadas para poder trabajar con elementos finitos. Entre las funcionalidades de este módulo se encuentra la posibilidad de modificar distintos parámetros (definición de longitud característica, tamaño de elemento mínimo, tamaño de elemento máximo, número mínimo de puntos para mallado de un elemento curvo, factor global sobre todas las longitudes características), modificar el algoritmo de mallado y hasta tratar de eliminar los nodos asociados a pocos elementos.

GMSH genera mallas no estructuras, y mallas estructuradas a partir de mallas no estructuradas (*transfinite* y *extrusion*). Los algoritmos no estructurados 2D generan triángulos o triángulos y cuadrángulos y los algoritmos no estructurados 3D generan tetraedros. Los algoritmos 2D estructuradas generan triángulos por defecto y los algoritmos 3D estructurados generan tetraedros, hexaedros, prismas y pirámides, en función del tipo de superficie.

#### EJEMPLO DE MALLADO 2D

Se necesita mallar el dominio en el que se resolverá la distribución de temperaturas en una sección de viga. Las dimensiones y nombres de bordes se presentan en la siguiente figura:



X	Y	Z				
0.0000	600.0000	0.0000				
0.0000	567.0000	0.0000				
97.0000	567.0000	0.0000				
97.0000	33.0000	0.0000				
0.0000	33.0000	0.0000				
0.0000	0.0000	0.0000				
215.0000	0.0000	0.0000				
215.0000	33.0000	0.0000				
118.0000	33.0000	0.0000				
118.0000	567.0000	0.0000				
215.0000	567.0000	0.0000				
215.0000	600.0000	0.0000				
0.0000	600.0000 0.000					



Se abre el programa GMSH y aparece un proyecto geométrico vacío por defecto denominado *untitled.geo*. Este proyecto puede ser salvado con un nombre adecuado (en este caso *ipn\_base.geo*).

File  $\rightarrow$  Save As



La incorporación del dominio a resolver en el proyecto geométrico se realiza a partir del ingreso de la entidad elemental básica.

Anternal and a Charles of M. States, Spin Street, game		Anona moa zacrizaria di ancigin dana 200	1000	Numerica Altarithm for 1946 Anna April Anna S	**
t Genkk tier ≣ manning an Physicing tier biologies		En Ener (b) Sam trib € Samore AA Dools Rada Sat Sat Sat Sat Sat Sat Sat Sat Sat			Month         R           Surgets         R           Surgets         R           Non         R           None         R           Science         Science           Surgets         Science
	Y Z_X		Y z_x		Y   <u>z_</u> x

### Elementary Entities $\rightarrow$ Add $\rightarrow$ New $\rightarrow$ Point

Existen varias posibilidades de ingreso de los puntos. La primera de ellas es vía la ventana emergente en donde se debe explicitar la coordenada del punto en las tres dimensiones, una longitud característica de mallado (*Characteristic Length*) y un espaciamiento de la grilla del dibujo (*Snapping Grid Spacing*). Otra de ellas permite ir ubicando los puntos con el cursor.

Los puntos que se generen para un mallado 2D tienen que tener la componente z=0, para que todo este en el mismo plano (plano z=0).

La longitud característica establece el tamaño del elemento relacionado con ese punto. La distribución de los tamaños de malla de elementos se obtiene por interpolación de estas longitudes características en toda la geometría.

El espaciamiento de la grilla del dibujo marca la ubicación de los posibles puntos de la malla en las tres dimensiones (si se elije [1] - [1] - [1], no es posible ubicar el punto [0.5,0.2,0.3]).



La opción restante tiene que ver con el código generado en el archivo ASCII *ipn\_base.geo*, en donde se caracteriza cada punto como:

### Point(i) = { $x_i$ , $y_i$ , $z_i$ , $lc_i$ };

En el caso del dominio a mallar los doce puntos de la tabla pueden ser ingresados directamente al archivo ASCII.

```
Point(1) = {0, 600, 0, 10};
Point(2) = {0, 567, 0, 10};
Point(3) = {97, 567, 0, 10};
Point(4) = {97, 33, 0, 10};
Point(5) = {0, 33, 0, 10};
Point(6) = {0, 0, 0, 10};
Point(7) = {215, 0, 0, 10};
Point(8) = {215, 33, 0, 10};
Point(9) = {118, 33, 0, 10};
Point(10) = {118, 567, 0, 10};
Point(11) = {215, 567, 0, 10};
```

Las longitudes características de la densidad de mallado también se pueden parametrizar. En el siguiente ejemplo se propone una mayor densidad de mallado en la zona cercana al Borde 4 y una menor en la zona del *Borde 2*. Todas las longitudes características que se propongan deben estar relacionadas por una relación de multiplicidad.

```
lc1 = 10;
lc2 = 20;
Point(1) = {0, 600, 0, lc1};
Point(2) = {0, 567, 0, lc1};
Point(3) = {97, 567, 0, lc1};
Point(4) = {97, 33, 0, lc2};
Point(5) = {0, 33, 0, lc2};
Point(6) = {0, 0, 0, lc2};
Point(6) = {215, 0, 0, lc2};
Point(8) = {215, 33, 0, lc2};
Point(9) = {118, 33, 0, lc2};
Point(10) = {118, 567, 0, lc1};
Point(11) = {215, 567, 0, lc1};
Point(12) = {215, 600, 0, lc1};
```

Los puntos incorporados al esquema geométrico pueden ser visualizados con numeraciones.



### Tools $\rightarrow$ Options $\rightarrow$ Geometry $\rightarrow$ Visibility

La unión de los puntos ya incorporados se realiza a través del agregado de las líneas.

 $\texttt{Elementary Entities} \rightarrow \texttt{Add} \rightarrow \texttt{New} \rightarrow \texttt{Straight Line}$ 

Cada línea se agrega seleccionando el inicio y el final del tramo. Para saber que el punto inicial o final del tramo está seleccionado, el cursor o el punto pueden cambiar de forma o color.



Luego de unir todos los puntos, si se abre el proyecto geométrico con un editor de texto se ve que se agregaron las líneas como:

```
Line(1) = {1, 12};
Line(2) = {12, 11};
Line(3) = {11, 10};
Line(4) = {10, 9};
Line(5) = {9, 8};
Line(6) = {8, 7};
Line(7) = {7, 6};
Line(8) = {6, 5};
Line(9) = {5, 4};
Line(10) = {4, 3};
Line(11) = {3, 2};
Line(12) = {2, 1};
```

Luego de haber cerrado el dominio con la entidad línea, se puede crear una superficie. En este caso, dado el problema bidimensional la superficie es plana (*Plane Surface*).

Elementary Entities  $\rightarrow$  Add  $\rightarrow$  New  $\rightarrow$  Plane Surface

Se selecciona con el cursor uno de los bordes de la superficie, y luego con 'e' se ejecuta automáticamente el *loop* que arma la superficie. En caso de que no cierre automáticamente el *loop* se deben seleccionar cada una de las líneas que cierran la superficie. Luego de crear la superficie, el modelo geométrico se completó.



## Line Loop(13) = {3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 1, 2}; Plane Surface(14) = {13};

Para mallar se debe ir al módulo Mesh. Eligiendo 2D se genera la malla bidimensional.



GMSH: Generador de Mallas

Mini Tutorial v1.1 - 19/Ene/2011 - Mariano Re

Aquí se muestran dos mallas obtenidas con dos longitudes características de los puntos distintas. Una malla más refinada se puede obtener modificando el archivo de proyecto \*.*geo* cambiando las longitudes características asociadas a los puntos, y también cambiando la longitud característica global en la interfaz gráfica:

Tools  $\rightarrow$  Options  $\rightarrow$  Mesh  $\rightarrow$  General  $\rightarrow$  Element size factor

N	/home/mre/Escritorio/IPN_base/ipn_bas	e_lc10.msh 👘 🔿 🗙	1	/home/mre/Escritorio/IPN_base/ipn_bas	e.geo	H (2)
		Gensh X Fle Tools Help Geometry X Elementary entries Mysick groups Edit Reload		XXXXXXXXX	Ele Tools Help Geometry Elementary embles Physical groups Edd Reload	
		₽   <u>7</u> x		XIXXXXXX		Y Z_X
XXYZC1:	1 @ S III III Geometry		XXYZC11	D S B C F B Geometry		

Al guardarse la malla ejecutada, se crea un archive ASCII del mismo nombre que el del proyecto geométrico pero de extensión *msh* y que tiene el siguiente formato:

```
$MeshFormat
```

```
2.1 0 8
$EndMeshFormat
$Nodes
130
1 0 600 0
2 0 567 0
. . .
129 107.5 355.4538461537227 0
130 107.5 388.3153846152803 0
$EndNodes
$Elements
270
1 15 3 0 1 0 1
2 15 3 0 2 0 2
. . .
11 15 3 0 11 0 11
12 15 3 0 12 0 12
13 1 3 0 1 0 1 13
14 1 3 0 1 0 13 14
. . .
103 1 3 0 11 0 92 2
104 1 3 0 12 0 2 1
GMSH: Generador de Mallas
```

Mini Tutorial v1.1 - 19/Ene/2011 - Mariano Re

105 2 3 0 14 0 9 65 49 106 2 3 0 14 0 9 4 65 ... 269 2 3 0 14 0 64 110 63 270 2 3 0 14 0 59 63 110 \$EndElements

Se presentan tres bloques de datos:

 $MeshFormat \rightarrow$  SendMeshFormat : Información del formato de malla

 $\text{SNodes} \rightarrow \text{SEndNodes}$  : Información del número global y ubicación de los nodos.

*\$Elements*  $\rightarrow$  *\$EndElements : Información de los elementos.* 

El bloque de información de los elementos contiene variada información por cada uno de ellos:

```
#elem tipo-elem #etq < etq1, etq2, etq3 > lista-nodos
```

(número de elemento, tipo de elemento, número de etiquetas, etiquetas, lista de nodos que pertenecen al elemento).

GMSH permite mallar con variados elementos, entre los más simples se encuentran puntos (tipo 15), líneas (tipo 1), triángulos (tipo 2) y cuadrángulos (tipo 3).



Por defecto, la primera etiqueta es el número de la entidad física a la que pertenece el elemento, la segunda es el número de la entidad geométrica elemental a la que pertenece el elemento, y la tercera es el número de particiones de la malla a la que pertenece el elemento, seguido de los identificadores de partición (un identificador negativo indica una celdas fantasmas).

Una forma de identificar ubicaciones de condiciones de borde es la de incorporar entidades físicas en el esquema geométrico (*Physical Groups*). Si se definen entidades físicas, el archivo salida \*.*msh* solamente contiene aquellos elementos que pertenecen a las mismas. Si se activa la casilla *save all*, se guardan todos los elementos. La identificación de a qué entidad física pertenece cierto elemento se da a través de una de las etiquetas en la definición del elemento.

#### Geometry $\rightarrow$ Physical Groups $\rightarrow$ Add $\rightarrow$ Line/Surface

En este ejemplo quedan definidos de la siguiente manera:

```
Physical Line("Borde 1") = {1};
Physical Line("Borde 2") = {2, 3, 4, 5, 6};
Physical Line("Borde 3") = {7};
Physical Line("Borde 4") = {8, 9, 10, 11, 12};
Physical Surface("Interior") = {18};
```

Luego se mallar, se encuentra un bloque más en donde se informa la cantidad y el nombre de las entidades físicas presentes. Los nodos y elementos presentados en el archivo de la malla son aquellos que pertenecen exclusivamente a las entidades físicas:

```
$MeshFormat
2.1 0 8
SEndMeshFormat
$PhysicalNames
5
1 1 "Borde 1"
1 2 "Borde 2"
1 3 "Borde 3"
1 4 "Borde 4"
2 5 "Interior"
$EndPhysicalNames
$Nodes
92
1 0 600 0
2 0 567 0
•••
91 48.49999999999791 567 0
92 24.2500000001908 567 0
$EndNodes
$Elements
92
1 1 3 1 1 0 1 13
2 1 3 1 1 0 13 14
•••
91 1 3 4 11 0 92 2
92 1 3 4 12 0 2 1
$EndElements
```

Por defecto, GMSH genera mallados no estructurados: triángulos en 2D y tetraedros en 3D. Estructurar la malla resulta posible, para esto se utiliza el algoritmo transfinito para obtener mallas de triángulos o tetraedros regulares para luego generar una malla de cuadrángulos o hexaedros mediante un comando de recombinación.

Una de las formas de obtener un mallado de cuadrángulos resulta a partir de la redefinición de de la geometría con el objetivo de subdividirla en varias superficies con cuatro extremos nodales. Para esto se incorporan nuevos puntos:



que en la codificación del archivo de geometría se describen como

```
Point(1) = \{0, 600, 0, 20\};\
Point(2) = \{0, 567, 0, 20\};\
Point(3) = \{97, 567, 0, 20\};\
Point(4) = \{97, 33, 0, 20\};\
Point(5) = \{0, 33, 0, 20\};\
Point(6) = \{0, 0, 0, 20\};
Point(7) = \{215, 0, 0, 20\};\
Point(8) = \{215, 33, 0, 20\};\
Point(9) = {118, 33, 0, 20};
Point(10) = \{118, 567, 0, 20\};
Point(11) = \{215, 567, 0, 20\};\
Point(12) = \{215, 600, 0, 20\};\
Point(13) = \{97, 0, 0, 20\};\
Point(14) = \{97, 600, 0, 20\};
Point(15) = \{118, 0, 0, 20\};\
Point(16) = \{118, 600, 0, 20\};
```

Luego de la redefinición de puntos, se unen los mismos con nuevas líneas:



Line(	1)	)	=	{	1,	,	1	4	}	;		
Line(	2)	)	=	{	14	ŀ,		1	6	}	;	
Line(	3)	)	=	{	16	5,		1	2	}	;	
Line(	4)	)	=	{	12	2,		1	1	}	;	
Line(	5)	)	=	{	11	.,		1	0	}	;	
Line(	6)	)	=	{	10	),		9	}	;		
Line(	7)	)	=	{	9,	,	8	}	;			
Line(	8)	)	=	{	8,	,	7	}	;			
Line(	9)	)	=	{	7,	,	1	5	}	;		
Line(	10	))	=		{1	.5	,		1	3	}	;
Line(	11	L)	=		{1	.3	,		6	}	;	
Line(	12	2)	=		{6	5,		5	}	;		
Line(	13	3)	=		{5	5,		4	}	;		
Line(	14	ł)	=		{4	Ŀ,		3	}	;		
Line(	15	5)	=		{3	3,		2	}	;		
Line(	16	5)	=		{2	2,		1	}	;		
Line(	17	7)	=		{1	.4	,		3	}	;	
Line(	18	3)	=		{3	3,		1	0	}	;	
Line(	19	))	=		{1	.0	,		1	6	}	;
Line(	20	))	=		{4	Ŀ,		1	3	}	;	
Line(	21	L)	=		{1	.5	,		9	}	;	
Line(	22	2)	=		٩	),		4	}	;		

Finalmente lo que se obtiene son siete superficies con cuatro extremos nodales:



```
Line Loop(23) = {1, 17, 15, 16};

Plane Surface(24) = {23};

Line Loop(25) = {2, -19, -18, -17};

Plane Surface(26) = {25};

Line Loop(27) = {3, 4, 5, 19};

Plane Surface(28) = {27};

Line Loop(29) = {7, 8, 9, 21};

Plane Surface(30) = {29};

Line Loop(31) = {22, 20, -10, 21};

Plane Surface(32) = {31};

Line Loop(33) = {13, 20, 11, 12};

Plane Surface(34) = {33};

Line Loop(35) = {18, 6, 22, 14};

Plane Surface(36) = {35};
```

A continuación se listan las líneas a discretizar con el algoritmo transfinito 1D. La expresión de la derecha indica el número de nodos que se crearán en esas líneas. En este ejemplo se indica la misma cantidad de nodos a aquellas líneas que están enfrentadas:

```
Transfinite Line {16,17,19,4}=5;
Transfinite Line {2,18}=3;
Transfinite Line {1,3,5,15}=5;
Transfinite Line {14,6}=20;
Transfinite Line {12,20,21,8}=5;
Transfinite Line {22,10}=3;
Transfinite Line {13,7,9,11}=5;
```

Existen dos argumentos opcionales en la sentencia *Transfinite*: *'Using Progression expression'* que indica que la distribución de nodos se haga según una progresión geométrica y *'Using Bump expression'* que indica que la distribución de nodos sea más refinada en los finales de cada línea.

El comando *Transfinite Surface* aplicará el algoritmo transfinito 2D a aquellas superficies que se identifiquen:



Transfinite Surface {24,26,28,30,32,34,36};

Con el comando *Recombine Surface* se recombinan las mallas triangulares de las superficies listadas para obtener mallas cuadrangulares.

Recombine Surface {24,26,28,30,32,34,36};

Λ	ipn_base_trans.geo	
		🚺 Gmsh 🗡
		<u>File</u> <u>Tools</u> <u>Help</u>
		Mesh
	Π	Define
	Π	Inspect
	H	Delete
	+	1D
	+	2D
		3D
	<u> </u>	First order
		Second order
		Refine by splitting
		Optimize
	H	Optimize (Netgen)
	+	Partition
	H	Reclassify
	4	Save
	T	
	+	
	+	
		V
		<u>Z</u> X

Con el problema resuelto, a partir de Las mallas obtenidas con estas metodologías, se tienen los resultados buscados en cada uno de los nodos en que se discretizó el dominio. Esa información puede ser incorporada al archivo de \*.msh y ser visualizada con el módulo *Post-processing* de GMSH. Para esto se incorpora el bloque *NodeData* que se detalla de la siguiente forma:

```
$NodeData
1
                     // 1 etiqueta string (título)
"Temperatura [*C]"
                     // etiqueta: el título de la vista
                     // 1 etiqueta real (tiempo)
1
0
                     // tiempo == 0
3
                     // 3 etiquetas integer
0
                     // paso temporal
                     // 1 campo componente (escalar)
1
364
                     // 364 valores (cantidad de nodos)
1
     20
2
     16
•••
362
     18.09636224
363
     18.52806235
364
     16.94692789
$EndNodeData
```



Respecto de las visualizaciones en la ventana de trabajo, con la rueda del mouse es posible acercarnos o alejarnos de nuestro esquema (la forma en que lo hace depende de la ubicación del cursor), con el botón derecho del mouse es posible mover rígidamente todo el esquema, y con el botón izquierdo es posible rotarlo. En el extremo inferior izquierdo de la ventana de trabajo se ubican una serie de opciones de visualización predeterminada (entre ellas *X*, *Y* y *Z* que clickeandolas ubica el dibujo en los planos respectivos, en este ejemplo *Z*).