





Notación:

-  Módulo principal del mét. numérico (preimplementado, no programable por el usuario).
-  Módulo del método numérico programable por el usuario.
-  Módulo del método numérico preimplementado (no programable por el usuario)
-  Modelo de datos (constantes, esquema numérico, etc.)

sample: variable de entrada a un módulo.

sample: variable de salida de un módulo.

dim1×dim2: dimensiones de una variable (matriz, vector o escalar)

Resumen de variables:

- **xnode**: matriz de pares (x,y) representando cada nodo de la malla.
- **icone**: matriz de conectividad. Todos los nodos se conectan formando elementos rectangulares (pero el tratamiento general del método es por nodos)
- **K**: matriz del sistema (difusión + reacción)
- **C**: matriz de masa (término reactivo/temporal).
- **F**: vector de flujo térmico.
- **PHI**: vector solución del método numérico (escalar)
- **DIR**: matriz de nodos frontera tipo Dirichlet.
- **NEU**: matriz de pares de nodos frontera tipo Neumann.
- **ROB**: matriz de pares de nodos frontera tipo Robin.
- **PUN**: matriz de nodos donde se aplican fuentes puntuales.
- **model**: struct con todos los datos del modelo (constantes, esquema numérico, etc.)
- **Q**: vector/matriz de flujo de calor.
- **kx**, **ky**: constantes de conductividad térmica del material en x e y, respectivamente.
- **c**: constante de reacción del material.
- **G**: fuente de calor. Es un vector que permite tener distintos valores para cada elemento.

Resumen de dimensiones de variables:

- **nnodes**: cantidad total de nodos de la malla.
- **nelem**: cantidad total de elementos de la malla.
- **nnod**: cantidad de nodos del elemento (3 o 4 nodos).
- **ndir**: cantidad de nodos frontera tipo Dirichlet.
- **nneu**: cantidad de pares de nodos (lado de un elemento) frontera tipo Neumann.
- **nrob**: cantidad de pares de nodos (lado de un elemento) frontera tipo Robin.
- **nrob**: cantidad de nodos frontera donde se aplican fuentes puntuales.
- **nit**: cantidad de iteraciones alcanzadas por el esquema temporal.



model

- `nnodes` [1x1 int] - Cantidad de nodos totales de la malla.
- `nelem` [1x1 int] - Cantidad de elementos totales de la malla.
- `kx` [1x1 double] - Conductividad térmica del material en sentido eje-x.
- `ky` [1x1 double] - Conductividad térmica del material en sentido eje-y.
- `c` [1x1 double] - Constante de reacción del sistema.
- `G` [nelemx1 double] - Fuente de calor volumétrica. Permite tener valores diferentes asociados a cada elemento de la malla.
- `ts` [1x1 int] - Parámetro de selección de esquema temporal: (0) Esquema Explícito, (1) Esquema Implícito y cualquier otro valor para estado estacionario.
- `rho` [1x1 double] - Densidad del material.
- `cp` [1x1 double] - Calor específico a presión constante del material.
- `maxit` [1x1 int] - Cantidad máxima de iteraciones (esquemas temporales).
- `tol` [1x1 double] - Tolerancia de error relativo entre iteraciones (esquemas temporales)
- `dt` [1x1 double] - Paso temporal de Esquema Implícito (arbitrario).
- `PHI_n` [nnodesx1 double] - Condición inicial para esquemas temporales. Es un vector que asigna un valor arbitrario inicial a cada nodo de la malla.

```
xnode [nnodes×2 double]
icone [nelem×4 int]
  DIR [ndir×1 double]
  NEU [nneu×1 double]
  ROB [nrob×1 double]
  PUN [npun×1 double]
model [1×1 struct]
```

fem2d_heat

```
PHI [nnodes×nit sparse double]
Q [nnodes×(2×nit) double]
```

```
% Inicialización de las variables principales del Sistema
1. [K,C,F] = fem2d_heat_initialize(model.nnodes);

% Ensamble de las matrices y vectores del sistema
2. [K,C,F] = fem2d_heat_gen_system(K,C,F,xnode,icone,model);

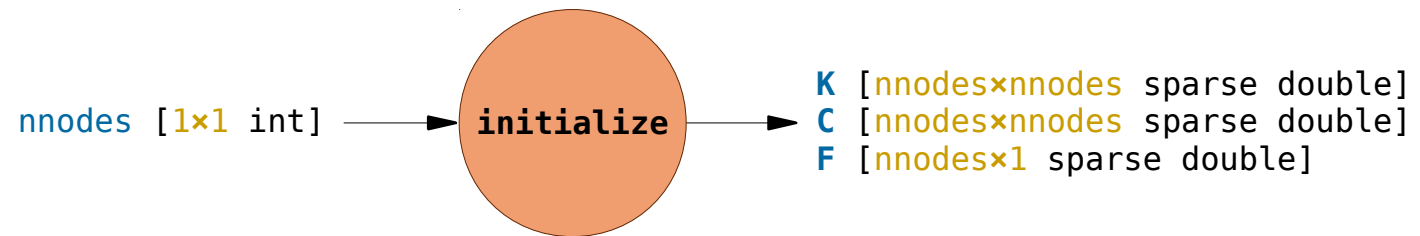
% Ensamble de nodos frontera Neumann
3. [F] = fem2d_heat_neumann(F,NEU,xnode);

% Ensamble de nodos frontera Robin
4. [K,F] = fem2d_heat_robin(KS,F,ROB,xnode);

% Ensamble de nodos con fuentes puntuales
5. [F] = fem2d_heat_pcond(F,xnode,icone,PUN);

% Ensamble de nodos frontera Dirichlet
6. [K,F] = fem2d_heat_dirichlet(K,F,DIR);

% Resolución del sistema lineal de ecuaciones
7. [PHI,Q] = fem2d_heat_solve(K,C,F,xnode,icone,model);
```



Descripción: módulo para inicializar las variables principales del sistema, las cuales se utilizarán para almacenar los datos calculados y ensamblados por el método numérico. Se inicializan como *sparse* para optimizar el rendimiento general del método numérico.

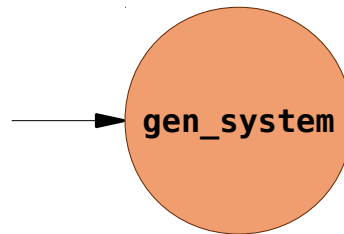
Entrada:

- `nnodes`: cantidad de nodos de la malla.

Salida:

- `K`: matriz del sistema (difusión + reacción)
- `C`: matriz de masa (término reactivo/esquemas temporales).
- `F`: vector de flujo térmico.

```
K [nnodes×nnodes double]
C [nnodes×nnodes double]
F [nnodes×1 double]
xnode [nnodes×2 double]
icone [nelem×4 int]
model [1×1 struct]
```



```
K [nnodes×nnodes sparse double]
C [nnodes×nnodes sparse double]
F [nnodes×1 sparse double]
```

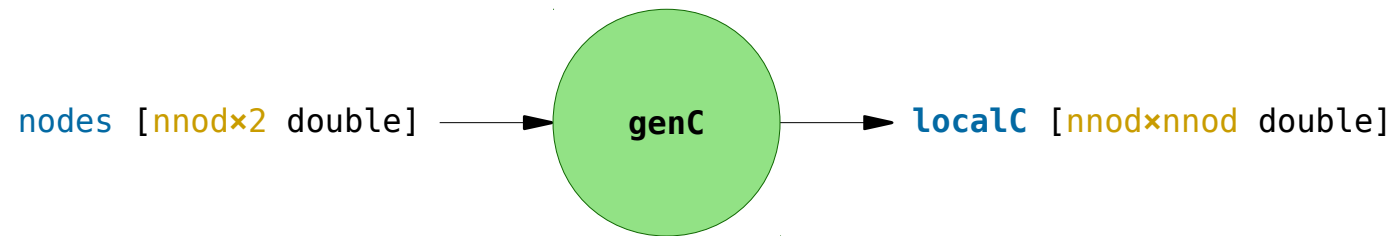
Descripción: módulo para realizar el ensamble de las variables globales del sistema. Es un *wrapper* con la estructura de bucle necesaria para llamar a los módulos que generan las matrices y vectores locales para cada elemento e ir ensamblándolas en las matrices y vectores globales del sistema (**K**, **C** y **F**).

Entrada:

- **K**: matriz global del sistema (difusión + reacción) vacía.
- **C**: matriz de masa global vacía.
- **F**: vector de flujo térmico global vacío.
- **xnode**: matriz de pares (x,y) representando cada nodo de la malla.
- **icone**: matriz de conectividad. Cada fila de la matriz indica la conectividad de un elemento triangular o rectangular con sus índices enumerados en sentido antihorario. Los elementos triangulares tienen la cuarta columna en -1.
- **model**: struct con todos los datos del modelo (constantes, esquema numérico, etc.)

Salida:

- **K**: matriz global del sistema (difusión + reacción) luego del ensamble.
- **C**: matriz de masa global luego del ensamble.
- **F**: vector de flujo térmico global luego del ensamble.



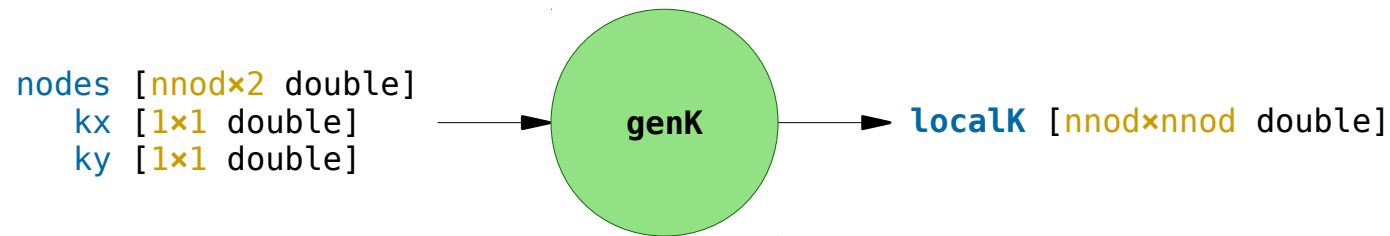
Descripción: módulo para calcular y evaluar de forma numérica la matriz de masa **C**. Se utilizan funciones de forma en coordenadas naturales y se resuelve la integral de forma numérica utilizando cuadratura de Gauss.

Entrada:

- **nodes**: nodos (x,y) del elemento. Los elementos admisibles son de 3 o 4 nodos.

Salida:

- **localC**: matriz de masa para elemento (local).



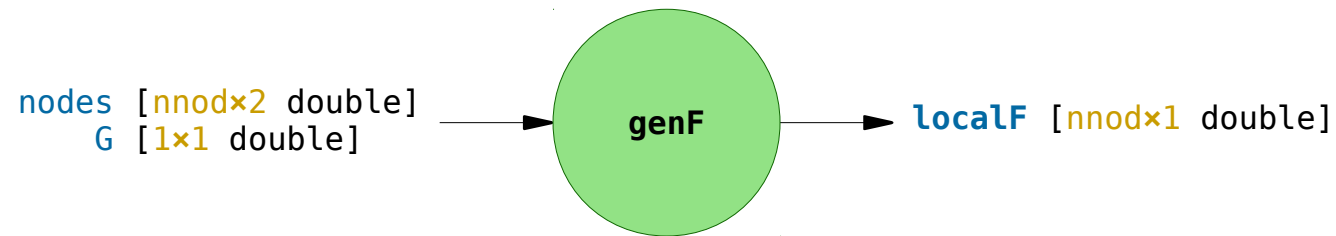
Descripción: módulo para calcular y evaluar de forma numérica la matriz de difusión K. Se utilizan funciones de forma en coordenadas naturales y se resuelve la integral de forma numérica utilizando cuadratura de Gauss.

Entrada:

- **nodes**: nodos (x,y) del elemento. Los elementos admisibles son de 3 o 4 nodos.
- **kx**: conductividad térmica orientada en eje-x.
- **ky**: conductividad térmica orientada en eje-y.

Salida:

- **localK**: matriz de difusión del elemento (local).



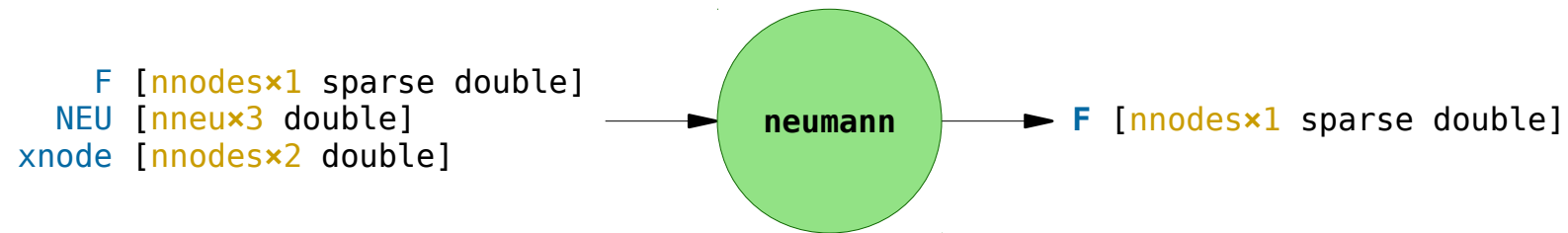
Descripción: módulo para calcular el vector de flujo térmico F para cada elemento, producto de la presencia de una fuente de calor en dicho elemento. La integral se resuelve mediante cuadratura de punto medio, y se requiere evaluar el área del elemento.

Entrada:

- `nodes`: nodos (x,y) del elemento. Los elementos admisibles son de 3 o 4 nodos.
- `G`: fuente de calor.

Salida:

- `localF`: vector de flujo térmico (local).



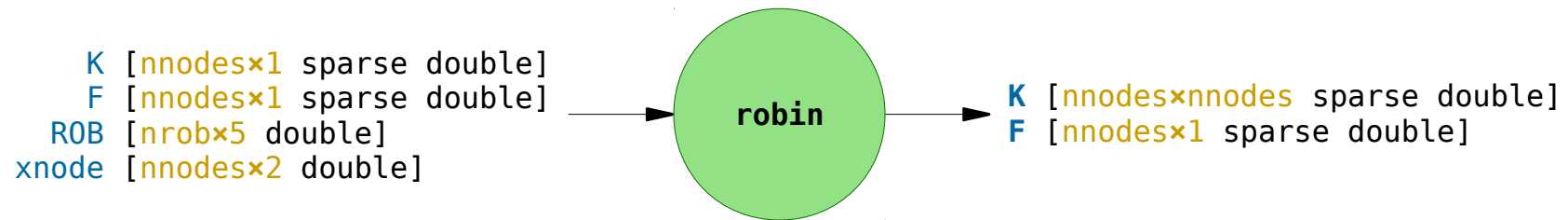
Descripción: módulo para calcular y ensamblar las contribuciones de nodos pertenecientes a fronteras de tipo Neumann.

Entrada:

- **F**: vector de flujo térmico.
- **NEU**: matriz con la información sobre la frontera de tipo Neumann.
 - Columnas 1-2: dos nodos contiguos formando un lado de un elemento.
 - Columna 3: valor de flujo térmico (q) asociado al lado del elemento.
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.

Salida:

- **F**: vector de flujo térmico. Presenta modificaciones luego de aplicar la condición de borde.



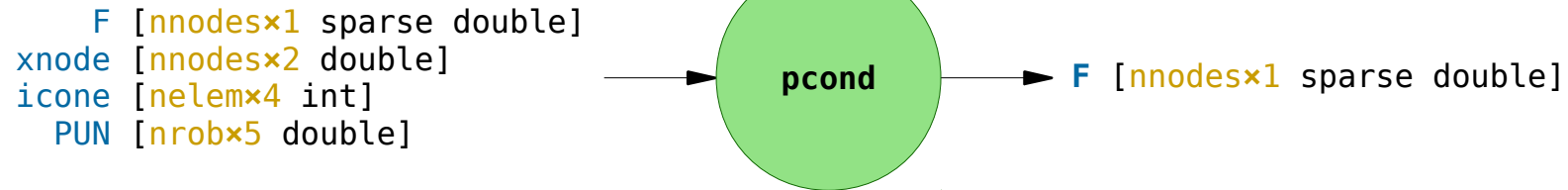
Descripción: módulo para calcular y ensamblar las contribuciones de nodos pertenecientes a fronteras de tipo Robin.

Entrada:

- **K**: matriz del sistema (difusión + reacción)
- **F**: vector de flujo térmico.
- **ROB**: matriz con la información sobre la frontera de tipo Robin.
 - Columnas 1-2: dos nodos contiguos formando un lado de un elemento.
 - Columna 3: valor de coeficiente de calor (**h**).
 - Columna 4: valor de temperatura de referencia (**phi_inf**).
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.

Salida:

- **K**: matriz del sistema (difusión + reacción) con modificaciones luego de aplicar la condición de borde.
- **F**: vector de flujo térmico con modificaciones luego de aplicar la condición de borde.



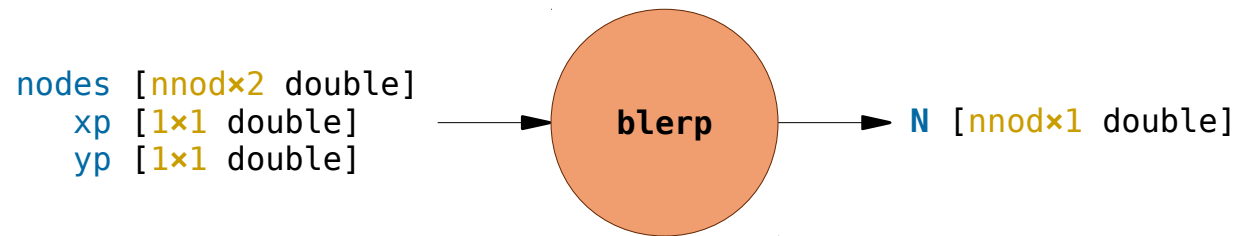
Descripción: módulo para calcular y ensamblar las contribuciones producidas por fuentes puntuales aplicadas en posiciones (x,y) dentro del dominio. Estos pares (x,y) también pueden ubicarse en los bordes del elemento, incluso en alguno de sus vértices.

Entrada:

- `F`: vector de flujo térmico.
- `xnode`: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- `icone`: matriz de conectividad. Indica los 3 ó 4 nodos que integran el elemento, recorridos en cualquier orden pero en sentido antihorario. En caso de elementos triangulares, la cuarta columna siempre es -1.
- `PUN`: matriz con la información sobre las fuentes puntuales aplicadas a elementos del dominio.
 - Columna 1: número de elemento al que se aplica la fuente.
 - Columna 2: valor de la fuente aplicada (`G`).
 - Columnas 3-4: posición absoluta (x,y) donde se aplica la fuente (cualquier parte del dominio).

Salida:

- `F`: vector de flujo térmico con modificaciones luego de aplicar la condición de borde.



Descripción: módulo para realizar una interpolación inversa dentro del elemento, a partir de conocer las coordenadas donde se aplica la interpolación. Se calculan las funciones de forma evaluadas en el punto de referencia (x_p, y_p). Para el caso de triángulos se aplica la definición de las funciones de forma y el cálculo es directo, mientras que para elementos cuadrangulares se debe realizar una interpolación bilineal inversa para obtener los parámetros (s, t) asociados al par (x_p, y_p).

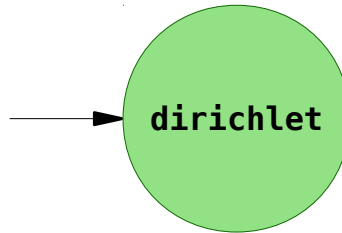
Entrada:

- **nodes**: nodos (x, y) del elemento. Los elementos admisibles son de 3 o 4 nodos.
- **xp**: coordenada- x del punto de referencia.
- **yp**: coordenada- y del punto de referencia.

Salida:

- **N**: funciones de forma para el elemento correspondientes al punto de referencia (x_p, y_p).

K [**nnodes**×**nnodes** sparse double]
F [**nnodes**×1 sparse double]
DIR [**ndir**×2 double]



K [**nnodes**×**nnodes** sparse double]
F [**nnodes**×1 sparse double]

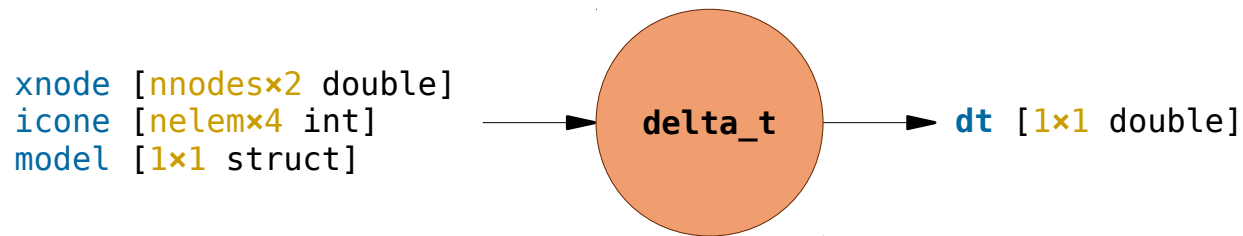
Descripción: módulo para calcular y ensamblar las contribuciones de nodos pertenecientes a fronteras de tipo Dirichlet.

Entrada:

- **K**: matriz del sistema (difusión + reacción)
- **F**: vector de flujo térmico.
- **DIR**: matriz con la información sobre la frontera de tipo Dirichlet.
 - Columna 1: número de nodo.
 - Columna 2: valor en ese nodo (escalar).

Salida:

- **K**: matriz del sistema (difusión + reacción) con modificaciones luego de aplicar la condición de borde.
- **F**: vector de flujo térmico con modificaciones luego de aplicar la condición de borde.



Descripción: módulo para calcular el paso temporal crítico para esquema temporal explícito a partir de las constantes del modelo y las dimensiones de los elementos de la malla.

Entrada:

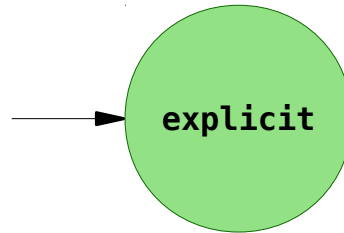
- `xnode`: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- `icone`: matriz de conectividad. Indica los 3 ó 4 nodos que integran el elemento, recorridos en cualquier orden pero en sentido antihorario. En caso de elementos triangulares, la cuarta columna siempre es -1.
- `model`: struct con todos los datos del modelo (constantes, esquema numérico, etc.)

Salida:

- `dt`: paso temporal crítico para método explícito.

fem2d_heat_explicit.m

```
K [nnodes×nnodes sparse double]
C [nnodes×nnodes sparse double]
F [nnodes×1 sparse double]
xnode [nnodes×2 double]
icone [nelems×4 int]
model [1×1 struct]
dt [1×1 double]
```



```
PHI [nnodes×(nit+1) sparse double]
Q [nnodes×(2×nit+1) double]
```

Descripción: módulo para resolver el sistema lineal de ecuaciones utilizando esquema temporal *explícito*. El primer valor (primer columna) es la condición inicial.

Entrada:

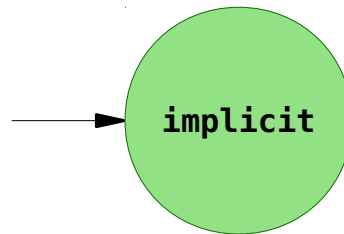
- **K**: matriz del sistema (difusión + reacción).
- **C**: matriz de masa del sistema (sin escalar por la constante de reacción).
- **F**: vector de flujo térmico.
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **icone**: matriz de conectividad. Indica los 3 ó 4 nodos que integran el elemento, recorridos en cualquier orden pero en sentido antihorario. En caso de elementos triangulares, la cuarta columna siempre es -1.
- **model**: struct con todos los datos del modelo (constantes, esquema numérico, etc.)
- **dt**: paso temporal *crítico* para método explícito.

Salida:

- **PHI**: matriz solución. Cada elemento del vector representa un valor escalar asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en **xnode**. Cada columna representa una iteración del esquema temporal (en total **nit** columnas).
- **Q**: matriz de flujo de calor. Para cada nodo se halla un vector bidimensional de flujo de calor, representado por un par (Q_x,Q_y). Cada par de columnas representa una iteración del esquema temporal (en total **2×nit** columnas).

fem2d_heat_implicit.m

```
K [nnodes×nnodes sparse double]
C [nnodes×nnodes sparse double]
F [nnodes×1 sparse double]
xnode [nnodes×2 double]
icone [nelems×4 int]
model [1×1 struct]
dt [1×1 double]
```



```
PHI [nnodes×(nit+1) sparse double]
Q [nnodes×(2×nit+1) double]
```

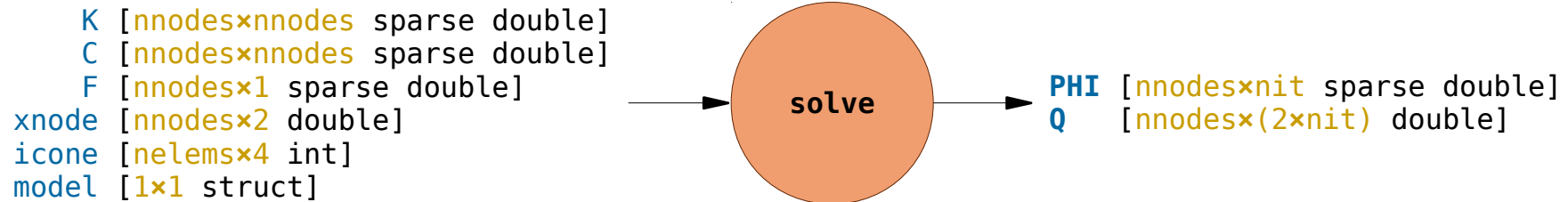
Descripción: módulo para resolver el sistema lineal de ecuaciones utilizando esquema temporal *implícito*. El primer valor (primer columna) es la condición inicial.

Entrada:

- **K**: matriz del sistema (difusión + reacción).
- **C**: matriz de masa del sistema (sin escalar por la constante de reacción).
- **F**: vector de flujo térmico.
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **icone**: matriz de conectividad. Indica los 3 ó 4 nodos que integran el elemento, recorridos en cualquier orden pero en sentido antihorario. En caso de elementos triangulares, la cuarta columna siempre es -1.
- **model**: struct con todos los datos del modelo (constantes, esquema numérico, etc.)
- **dt**: paso temporal *arbitrario* para método implícito.

Salida:

- **PHI**: matriz solución. Cada elemento del vector representa un valor escalar asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en **xnode**. Cada columna representa una iteración del esquema temporal (en total **nit** columnas).
- **Q**: matriz de flujo de calor. Para cada nodo se halla un vector bidimensional de flujo de calor, representado por un par (Q_x,Q_y). Cada par de columnas representa una iteración del esquema temporal (en total **2×nit** columnas).



Descripción: módulo para resolver el sistema lineal de ecuaciones. En este módulo se realizan los cálculos para obtener la solución propia del método numérico. Dicha solución se obtiene por dos vías:

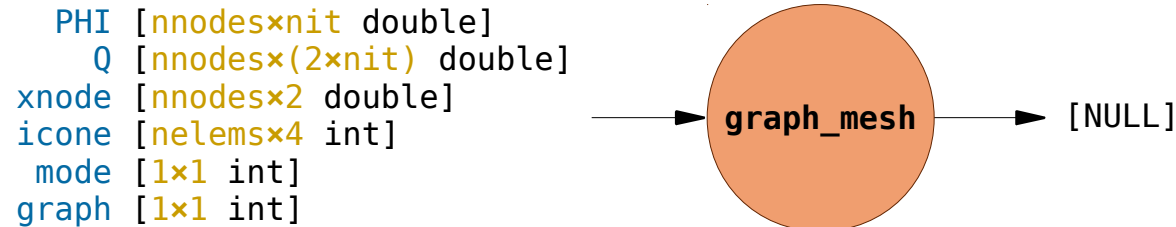
- Sin la aplicación de esquemas temporales, es decir, la solución del sistema en estado estacionario. Resolución por método directo.
- Aplicación de esquemas temporales, a saber: método explícito y método implícito. Se evalúa la evolución temporal del sistema desde un estado inicial conocido hasta un determinado instante de tiempo. Resolución por método iterativo.

Entrada:

- **K**: matriz del sistema (difusión + reacción).
- **C**: matriz de masa del sistema (sin escalar por la constante de reacción).
- **F**: vector de flujo térmico.
- **xnode**: matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **icone**: matriz de conectividad. Indica los 3 ó 4 nodos que integran el elemento.
- **model**: struct con todos los datos del modelo (constantes, esquema numérico, etc.)

Salida:

- **PHI**: matriz solución. Cada elemento del vector representa un valor escalar asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en **xnode**. Cada columna representa una iteración del esquema temporal (en total **nit** columnas).
- **Q**: matriz de flujo de calor. Para cada nodo se halla un vector bidimensional de flujo de calor, representado por un par (Qx,Qy). Cada par de columnas representa una iteración del esquema temporal (en total **2×nit** columnas).

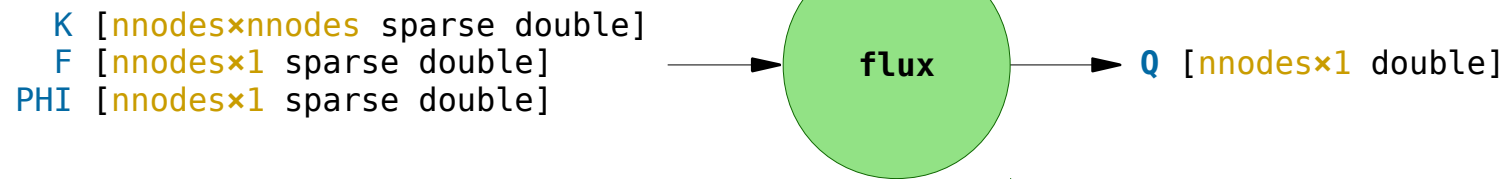


Descripción: módulo para graficar la solución del método numérico. Posee distintas formas de operación y diferentes tipos de gráficas.

Entrada:

- **PHI:** matriz solución. Cada elemento del vector representa un valor escalar asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en **xnode**. Cada columna representa una iteración del esquema temporal (en total **nit** columnas).
- **Q:** matriz de flujo de calor. Para cada nodo se halla un vector bidimensional de flujo de calor, representado por un par (Qx,Qy). Cada par de columnas representa una iteración del esquema temporal (en total **2*nit** columnas).
- **xnode:** matriz de nodos con pares (x,y) representando las coordenadas de cada nodo de la malla.
- **icone:** matriz de conectividad. Cada fila de la matriz indica la conectividad de un elemento rectangular, comenzando por el extremo inferior izquierdo y recorriendo el elemento en sentido antihorario.
- **mode:** modo de visualización: 2D, 3D, con o sin malla.
- **graph:** tipo de gráfica:
 - Temperatura (escalar)
 - Flujo de calor (vectorial)
 - Flujo eje-x (escalar)
 - Flujo eje-y (escalar)
 - Magnitud de flujo (escalar).

Salida: ninguna.



Descripción: módulo calcular el flujo de calor en todo el dominio. Se aplica la Ley de Fourier y se evalúa como fluye el calor en todos los puntos (nodos) del dominio.

Entrada:

- `K`: matriz del sistema (difusión + reacción)
- `F`: vector de flujo térmico.
- `PHI`: vector solución. Cada elemento del vector representa un valor escalar asociado a cada nodo de la malla, y su posición dentro del vector depende de cómo se especificó cada nodo en `xnode`.

Salida:

- `Q`: vector de flujo de calor.