

MODELADO Y OPTIMIZACIÓN DE SISTEMAS DE PARÁMETROS DISTRIBUIDOS ECOSIMPRO

Carmen G. Moles, Antonio A. Alonso y Julio R. Banga*

Process Engineering Group, IIM-CSIC
c/ Eduardo Cabello 6, 36208, Vigo, Spain

Resumen

En este trabajo se detallan los pasos seguidos para poder modelar sistemas de parámetros distribuidos en EcosimPro. Para ello, se ha implementado mediante macros el método NMOL para una y dos dimensiones espaciales. Asimismo, se ha explotado la estructura dispersa del conjunto de DAEs resultante tras la discretización espacial, lo que ha permitido la reducción de los tiempos de computación necesarios en las simulaciones de sistemas descritos por ecuaciones diferenciales en derivadas parciales.

Palabras Clave: EcosimPro, simulación dinámica, ecuaciones diferenciales parciales, sistemas distribuidos.

1 INTRODUCCION

EcosimPro® es un entorno de simulación con un lenguaje de alto nivel (EL) que permite declarar sistemas descritos mediante DAEs e incluso eventos discretos de un modo sencillo e intuitivo. Sin embargo, no permite declarar directamente sistemas formulados mediante ecuaciones diferenciales en derivadas parciales (PDEs, o en general, PDAEs).

Por este motivo en este trabajo presentamos la implementación del método numérico de las líneas (NMOL, numerical method of lines) para una y dos dimensiones en el entorno de EcosimPro (para más detalles sobre el método NMOL consultar los libros de Schiesser [6], [7]).

Como primera aproximación desarrollamos el método NMOL mediante la definición de funciones con el lenguaje EL de EcosimPro. A pesar del buen comportamiento observado en los sistemas distribuidos implementados, el tiempo de computación necesario se incrementaba rápidamente con la dimensión del sistema.

Para superar esta dificultad, se incorporaron macros, en sustitución de las funciones, lo que ha permitido una considerable reducción del tiempo de computación de cada simulación. Ello es debido, fundamentalmente, a que mediante las macros, EcosimPro tiene acceso a toda la estructura simbólica

de las ecuaciones DAEs resultantes de la discretización NMOL, lo que permite explotarla adecuadamente mediante integradores que utilicen álgebra dispersa (sparse).

En las siguientes secciones se introducen brevemente los fundamentos del método NMOL y se muestra su implementación y uso en EcosimPro. Finalmente, mediante dos ejemplos, se muestran las ventajas del uso de macros.

2 DESCRIPCION DEL METODO NUMERICO DE LAS LINEAS

El método numérico de las líneas (numerical method of lines, o NMOL) [6] es una técnica para la resolución de ecuaciones en derivadas parciales (PDEs) de tipo parabólico o hiperbólico a través de su transformación en un conjunto de ODEs o DAEs que puede ser resuelto con un integrador estándar. Se basa en la discretización del dominio espacial utilizando distintas metodologías que emplean mallas uniformes, no uniformes o adaptativas [3], [4].

El valor de la variable dependiente en un punto de la malla se calcula a partir de los valores en los puntos adyacentes, siendo el número de puntos utilizados el que determina el tipo u orden del método.

Con el fin de ilustrar el proceso de obtención de estas fórmulas, se detallan los pasos seguidos para la obtención de la fórmula de orden 2 (más detalles en los libros de Schiesser [7], [8]) con una malla uniforme en el espacio.

Supóngase un problema unidimensional (los resultados son fácilmente extrapolables a casos 2D o 3D) $x=x(\xi)$. Se considerará que una buena aproximación de la solución a la PDE en un punto viene dada por el desarrollo en serie de Taylor:

$$x(\mathbf{x}) = a_0 + a_1(\mathbf{x} - \mathbf{x}_i) + a_2(\mathbf{x} - \mathbf{x}_i)^2 + a_3(\mathbf{x} - \mathbf{x}_i)^3 + \dots \quad (1)$$

siendo:

$$a_n = \frac{1}{n!} \frac{\partial^n x(\mathbf{x}_i)}{\partial \mathbf{x}^n} \quad (2)$$

Si se utiliza una malla de σ elementos equidistantes y utilizando la expresión para calcular el valor del campo en dos puntos no consecutivos de la malla se obtiene:

$$\begin{aligned} x(\mathbf{x}_{i-1}) &= x(\mathbf{x}_i) + \frac{\partial x(\mathbf{x}_i)}{\partial \mathbf{x}} (\mathbf{x}_{i-1} - \mathbf{x}_i) + \frac{1}{2!} \frac{\partial^2 x(\mathbf{x}_i)}{\partial \mathbf{x}^2} (\mathbf{x}_{i-1} - \mathbf{x}_i)^2 + \dots \\ x(\mathbf{x}_{i+1}) &= x(\mathbf{x}_i) + \frac{\partial x(\mathbf{x}_i)}{\partial \mathbf{x}} (\mathbf{x}_{i+1} - \mathbf{x}_i) + \frac{1}{2!} \frac{\partial^2 x(\mathbf{x}_i)}{\partial \mathbf{x}^2} (\mathbf{x}_{i+1} - \mathbf{x}_i)^2 + \dots \end{aligned} \quad (3)$$

y restando las dos expresiones anteriores resulta:

$$\frac{\partial x(\mathbf{x}_i)}{\partial \mathbf{x}} = \frac{x(\mathbf{x}_{i+1}) - x(\mathbf{x}_{i-1})}{2\Delta \mathbf{x}} + O(\Delta \mathbf{x}^2) \quad (4)$$

De esta manera es posible calcular directamente el valor de la derivada espacial en función de los valores del campo en dos puntos no consecutivos de la malla. Sin embargo, esta expresión es válida únicamente para $i=2, \dots, \sigma-1$. Para el primer y último puntos de la malla habrá que calcular otras expresiones, y así para aproximar el valor de la derivada espacial en el punto ξ_1 se utilizarán los valores del campo en los puntos ξ_2 y ξ_3 :

$$\begin{aligned} x(\mathbf{x}_2) &= x(\mathbf{x}_1) + \frac{\partial x(\mathbf{x}_1)}{\partial \mathbf{x}} \Delta \mathbf{x} + \frac{1}{2!} \frac{\partial^2 x(\mathbf{x}_1)}{\partial \mathbf{x}^2} (\Delta \mathbf{x})^2 + \dots \\ x(\mathbf{x}_3) &= x(\mathbf{x}_2) + \frac{\partial x(\mathbf{x}_2)}{\partial \mathbf{x}} \Delta \mathbf{x} + \frac{1}{2!} \frac{\partial^2 x(\mathbf{x}_2)}{\partial \mathbf{x}^2} (\Delta \mathbf{x})^2 + \dots \end{aligned} \quad (5)$$

y restando:

$$\frac{\partial x(\mathbf{x}_1)}{\partial \mathbf{x}} = \frac{-3x(\mathbf{x}_1) + 4x(\mathbf{x}_2) - x(\mathbf{x}_3)}{2\Delta \mathbf{x}} + O(\Delta \mathbf{x}^2) \quad (6)$$

De igual manera calculando las aproximaciones para $\xi_{\sigma-2}$ y $\xi_{\sigma-1}$ y restando, se obtiene la siguiente expresión:

$$\frac{\partial x(\mathbf{x}_p)}{\partial \mathbf{x}} = \frac{3x(\mathbf{x}_p) - 4x(\mathbf{x}_{p-1}) + x(\mathbf{x}_{p-2})}{2\Delta \mathbf{x}} + O(\Delta \mathbf{x}^2) \quad (7)$$

Ahora se dispone de una aproximación de la primera derivada espacial para cada punto de la malla. El desarrollo de las fórmulas para otros órdenes sigue un procedimiento similar.

Las fórmulas de segundo orden (de 3 puntos) expresadas de un modo compacto serían:

$$\frac{d\tilde{x}}{d\mathbf{x}} = \frac{1}{2\Delta \mathbf{x}} \begin{bmatrix} -3 & 4 & -1 \\ -1 & 0 & 1 \\ 1 & -4 & 3 \end{bmatrix} \tilde{x} \quad (8)$$

El cálculo de las derivadas espaciales de segundo orden se puede realizar aplicando recursivamente estas fórmulas. Al sustituir directamente en la PDE original se obtiene un conjunto de ODEs, una por cada nodo de la malla. El número de ODEs resultante depende por tanto del nivel de discretización espacial elegido, aumentando de manera notable a medida que aumenta la no linealidad y la dimensión espacial del problema.

En este trabajo se ha optado por no utilizar recursivamente las fórmulas de primer orden para el cálculo de las derivadas espaciales de segundo orden. Se ha implementado el método de las líneas utilizando las fórmulas de segundo orden (3 puntos) tanto para el cálculo de la derivada espacial de primer orden como de segundo, cuyas fórmulas se obtienen fácilmente tras un desarrollo similar al presentado anteriormente.

3 IMPLEMENTACION DE MACROS EN ECOSIMPRO

Una macro es un grupo de instrucciones en un programa que se codifican solo una vez, asignándoles un nuevo nombre, y que pueden utilizarse cuantas veces sea necesario (y de forma recursiva). En la ejecución, cada macro es sustituida por el grupo de instrucciones que codifica, actuando sobre posibles parámetros definidos como argumentos.

Las macros de EcosimPro han sido implementadas por P. Cobas (EA Int.) al explicarle las necesidades que teníamos para la implementación del método NMOL. El uso de macros implica un preprocesado del fichero que contiene el componente (ej. llamado fichero.el). En este preprocesado se expanden las macros, definidas en un archivo llamado macros.h, creando un nuevo fichero EL (ej. nuevofichero.el) que es preciso compilar en un segundo paso, del modo usual. Tras este preprocesado la estructura de las ecuaciones (DAEs) generadas por el método NMOL, y en particular sus propiedades dispersas, están ya disponibles para ser explotadas adecuadamente por integradores tipo sparse.

Para la validación de las macros en el entorno de EcosimPro se utilizaron varios modelos previamente validados con datos experimentales reales. Se estableció el valor de los parámetros del modelo y se almacenaron adecuadamente los valores de salida para su posterior representación y análisis.

4 DEFINICION DE MACROS

Las macros se declaran en un archivo cabecera de C++, (por ejemplo llamado macros.h). Nota: para poder usar estas macros es preciso sustituir una de las librerías de EcosimPro. Contactar con Pedro Cobas para más información

- Este fichero se introduce en un directorio accesible por EcosimPro.

El único requisito para su utilización es incluir este fichero en el comienzo del componente mediante:

```
#include<macro dir>/ macros.h
```

Como ejemplo, introducimos el código que implementa el método NMOL en una dimensión cuando no se especifican condiciones frontera:

```
// Crea el termino dfx para las diferencias finitas:
```

```
#define dfx(xL,xU,N)\
(1.0 / (2.0*((xU-xL) / (N -1))))
```

```
// Calcula la primera derivada de  $\xi$  en  $\xi_d$  sin imponer condiciones extremo:
```

```
#define PDE_1D(xL,xU, N,  $\xi$ ,  $\xi_d$ )\
```

```
 $\xi_d[1]=dfx(xL,xU,N)*(-3*\xi[1]+4.0*\xi[2]-1.0*\xi[3]) \&\$ 
```

```
EXPAND (i IN 2,N-1) &\
```

```
 $\xi_d[i]=dfx(xL,xU,N)*(-\xi[i-1] + \xi[i+1]) \&\$ 
```

```
 $\xi_d[N]=dfx(xL,xU,N)*(\xi[N-2]-4.0*\xi[N-1]+ 3.0*\xi[N])$ 
```

donde xL es el límite inferior de x (dimensión), xU es el límite superior de x, N el número de intervalos del dominio (incluyendo los puntos frontera), ξ la variable a diferenciar y ξ_d la primera derivada.

5 FORMULACION DEL METODO NMOL PARA UNA DIMENSION EN ECOSIMPRO

Al implementar el método NMOL en EcosimPro hemos considerado la existencia de diferentes tipos de condiciones frontera tales como Dirichlet y Neumann, incluso que tales condiciones frontera no se especifiquen. Para cada caso, la llamada al método NMOL desde el correspondiente componente de EcosimPro sufre pequeñas variaciones tales como:

· *Cálculo de la primera derivada de \mathbf{x} en x (dimensión) sin imponer condiciones extremo:*

```
PDE_1D(xL, xU, N,  $\xi$ ,  $\xi_d$ )
```

donde xL es el límite inferior de x, xU es el límite superior de x, N el número de intervalos del dominio (incluyendo los puntos frontera), ξ la variable a diferenciar y ξ_d la primera derivada.

· *Cálculo de la primera y segunda derivada de \mathbf{x} sin imponer condiciones extremo:*

```
PDE_1D_2der(xL, xU, N,  $\xi$ ,  $\xi_d$ ,  $\xi_{dd}$ )
```

donde ξ_{dd} es la segunda derivada de la variable ξ .

· *Cálculo de la primera derivada de \mathbf{x} imponiendo condiciones extremo:*

```
PDE_1D_EXTR(xL, xU, N,  $\xi$ ,  $\xi_d$ , flag1, ux1, flagN, uxN)
```

donde flag1 indica si existen condiciones frontera de segundo orden (Neumann) en el borde 1 (TRUE en caso afirmativo, FALSE en caso negativo) y flagN si existen condiciones frontera de segundo orden en el borde N.

Además, ux1 es el valor de la condición frontera en el borde 1 y uxN en el borde N.

· *Cálculo de la primera y segunda derivada de \mathbf{x} imponiendo condiciones extremo:*

```
PDE_1D_EXTR_2der(xL, xU, N,  $\xi$ ,  $\xi_d$ ,  $\xi_{dd}$ , flag1, ux1, flagN, uxN)
```

En una única macro se encuentran definidos los diferentes casos, facilitando el uso de la misma.

6 FORMULACION DEL METODO NMOL PARA DOS DIMENSIONES EN ECOSIMPRO

Al igual que en una dimensión se han implementado diferentes llamadas al método NMOL dependiendo de las condiciones frontera existentes. La principal diferencia respecto a lo descrito para una dimensión es que en este caso también debe especificarse la dimensión respecto a la cual se deriva. Las diferentes posibilidades son:

· *Cálculo de la primera derivada de ξ en x (dimensión) sin imponer condiciones extremo:*

```
PDE_2D(xL, xU, N1, N2, flag,  $\xi$ ,  $\xi_d$ )
```

donde xL es el límite inferior de x, xU es el límite superior de x, N1 el número de intervalos del dominio 1 (incluyendo los puntos frontera), N2 del dominio 2, flag indica la dimensión respecto a la cual se deriva (TRUE indica primera dimensión y FALSE segunda

dimensión), ξ es la variable a diferenciar y ξ_d la primera derivada.

· Cálculo de la primera y segunda derivada de \mathbf{x} en x sin imponer condiciones extremo:

PDE_2D_2der(xL, xU, N1, N2, flag, ξ , ξ_d , ξ_{dd})

donde ξ_{dd} es la segunda derivada calculada numéricamente.

· Cálculo de la primera derivada de \mathbf{x} imponiendo condiciones extremo:

PDE_2D_EXTR(xL, xU, N1, N2, flag, ξ , ξ_d , flag1, ux1, flagN, uxN)

donde flag1 indica si existen condiciones frontera de segundo orden en el borde 1 y flagN si existen condiciones frontera de segundo orden en el borde N1 o N2 dependiendo de la dimensión respecto a la cual se derive.

· Cálculo de la primera y segunda derivada de ξ en x imponiendo condiciones extremo:

PDE_2D_EXTR_2der(xL, xU, N1, N2, flag, ξ , ξ_d , ξ_{dd} , flag1, ux1, flagN, uxN)

En cada caso, y dependiendo de las características del problema (si tiene condiciones frontera y de que orden), se ejecutará la llamada a una u otra función.

7 EJEMPLOS

Un gran número de los procesos que tienen lugar en la industria alimentaria y biotecnológica se caracterizan por formularse mediante ecuaciones diferenciales en derivadas parciales.

Se han seleccionado dos ejemplos correspondientes respectivamente a la esterilización de alimentos en envases cilíndricos y al procesamiento aséptico de medios líquidos para mostrar como sería la implementación de cualquier sistema disperso mediante macros en EcosimPro. Por otro lado, se exponen las ventajas que presenta la implementación del método NMOL mediante macros respecto al uso de funciones.

7.1 ESTERILIZACION DE ALIMENTOS EN UN ENVASE CILINDRICO (PROBLEMA BIDIMENSIONAL)

Se considera la esterilización de alimentos sólidos o líquidos muy viscosos (mecanismo de calentamiento exclusivamente por conducción) que se encuentran preenvasados. El modelo matemático se ha tomado de Banga y co. [2] y consta de:

· Ecuación de transmisión de calor en estado no estacionario (Fourier)

$$\frac{dT}{dt} = \mathbf{a} \left(\frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial z^2} \right) \quad (9)$$

$$\frac{dF_c}{dt} = 10^{(T(0,0) - T_{microorg}^*) / Z_{microorg}}$$

siendo T la temperatura del alimento en el interior del recipiente.

· Cinéticas de muerte térmica de esporas de microorganismos y cinéticas de degradación térmica de nutrientes y/o factores organolépticos

$$\frac{dX(r, z)}{dt} = - \frac{2.3025}{D_{microorg}^*} X(r, z) \cdot e^{2.3025(T(r, z) - T_{microorg}^*) / Z_{microorg}}$$

$$\frac{dS(r, z)}{dt} = - \frac{2.3025}{D_{nutrient}^*} S(r, z) \cdot e^{2.3025(T(r, z) - T_{nutrient}^*) / Z_{nutrient}}$$

siendo X y S la concentración de microorganismo y nutriente respectivamente.

· Evaluación de la letalidad y concentración media final de nutriente y microorganismo

$$X_{media} = \frac{2}{V_t} \int_0^R \int_0^L r X(r, z) dr dz \quad (11)$$

$$S_{media} = \frac{2}{V_t} \int_0^R \int_0^L r S(r, z) dr dz$$

$$F_s = - D_{microorg}^* \log\left(\frac{X_{media}}{X_0}\right) \quad (12)$$

siendo F_c la letalidad en el punto crítico y F_s la letalidad total, integrada a toda la lata. Finalmente X_{media} y S_{media} son los valores medios de concentración de microorganismo y nutriente.

Además, para definir correctamente el modelo se fijan como condiciones frontera,

$$\begin{aligned} \frac{\partial T(r, L)}{\partial z} &= T_{retort} \\ \frac{\partial T(R, z)}{\partial r} &= T_{retort} \\ \frac{\partial T(0, z)}{\partial r} &= 0 \\ \frac{\partial T(r, 0)}{\partial z} &= 0 \end{aligned} \quad (13)$$

Con el fin de ilustrar la implementación del componente se incluye un extracto del código EL correspondiente al modelo de esterilización, centrándonos en la implementación de la ecuación de transmisión del calor en estado no estacionario y el cálculo de las derivadas parciales:

```
#include "\macros.h"
COMPONENT EstCilind

CONTINUOUS

-- Derivada 1ª y 2ª respecto a r:
-- Condiciones frontera:

EXPAND(i IN 1,NR) N1UxN[i] = 0
EXPAND(i IN 1,NR) N1Ux1[i] = 0

-- Llamada a NMOL (macro):
PDE_2D_EXTR_2der(0,R,NR,NL,TRUE,
T,Tr,Trr,TRUE,N1Ux1,FALSE,N1Ux)

-- Derivada 1ª y 2ª respecto a z:
-- Condiciones frontera:

EXPAND(i IN 1,NL) N2UxN[i] = 0
EXPAND(i IN 1,NL) N2Ux1[i] = 0

-- Llamada a NMOL (macro):
PDE_2D_EXTR_2der(0,L,NR,NL,FALSE,
T,Tz,Tzz,TRUE,N2Ux1,FALSE,N2UxN)

-- Ley de Fourier:
EXPAND(j IN 1,NL-1)
T[1,j]'= alfa*(2.0*Trr[1,j] + Tzz[1,j])

EXPAND(i IN 2,NR-1)
EXPAND(j IN 1,NL-1)
T[i,j]'=alfa*(Trr[i,j]+((1/((i-1)*DR))*
Tr[i,j]) +Tzz[i,j])
```

Tras expandir las ecuaciones diferenciales mediante el método NMOL el sistema queda formado por:

Tabla 1: Descripción del modelo

Ecuaciones	V. totales	V. explícitas	V. dinámicas
1462	1462	1119	343

Las figuras 1 y 2 muestran que la simulación del proceso térmico mediante las funciones originales (color azul) y las nuevas MACROS (color rojo) conduce exactamente a los mismos resultados.

Figura 1: Representación de la letalidad en el punto crítico F_c y total F_s (integrada a toda la lata)

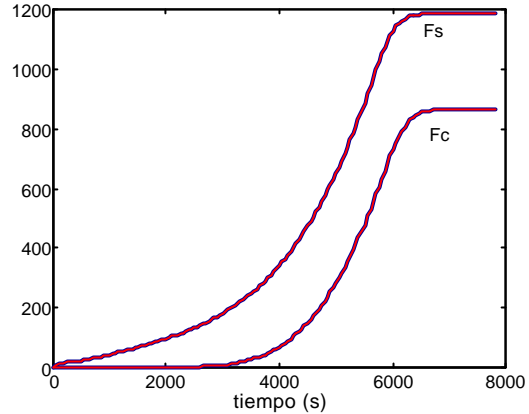
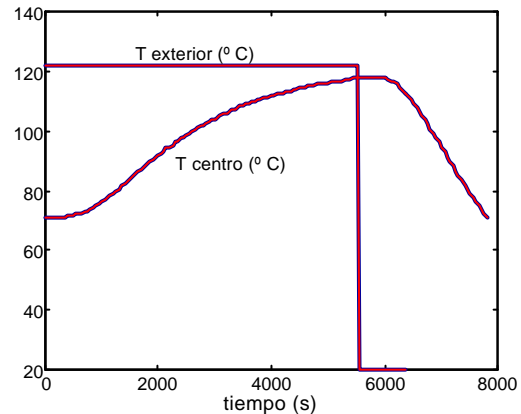


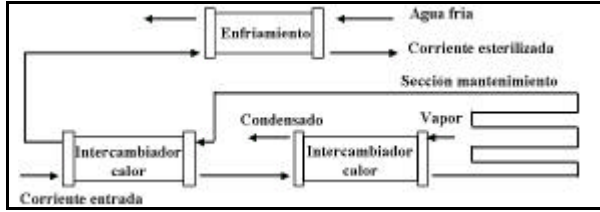
Figura 2: Evolución de la temperatura en el exterior y centro de la lata cilíndrica



7.2 PROCESAMIENTO ASEPTICO

La operación considerada en este segundo ejemplo es la simulación de un procesamiento aseptico. El esterilizador continuo está formado por dos intercambiadores de calor y una sección de mantenimiento (ver figura 3) basándonos en el modelo propuesto por Armenante y Leskowicz [1]. En el primer intercambiador, la corriente de alimentación sufre un precalentamiento por contacto indirecto con la corriente estéril que sale de la sección de mantenimiento (ahorro de energía). En el segundo intercambiador de calor, la corriente precalentada alcanza la temperatura de esterilización por contacto indirecto con vapor. Finalmente, en la sección de mantenimiento se mantiene esta temperatura de esterilización el tiempo necesario para obtener la reducción de microorganismos deseada

Figura 3: Esquema del proceso aseptico



La temperatura en el medio fluido es función del tiempo mientras que en las partículas esféricas también lo es de la posición.

Como microorganismo contaminante de referencia, se consideró el *Bacillus stearotherophilus* debido a su resistencia térmica a la esterilización (fijando un grado de esterilización seguro). Esta reducción debe alcanzarse en la corriente líquida únicamente por la contribución de la sección de mantenimiento (aunque realmente existe una contribución global de los demás intercambiadores de calor). El modelo ha sido implementado en EcosimPro haciendo uso del método NMOL implementado mediante macros. Finalmente se obtiene un sistema formado por 2955 ecuaciones y 672 variables dinámicas (para más detalles ver tabla 2).

Tabla 2: Descripción del modelo

Ecuaciones	V. totales	V. explícitas	V. dinámicas
2955	2955	2283	672

Debido a la extensión de este modelo solamente incluimos los balances de materia y energía de la fase líquida y sólida en la sección de mantenimiento, siendo:

SECCION MANTENIMIENTO (Medio Líquido)

Balance materia:

$$\frac{4v_f}{\rho D^2} \frac{\partial \log N(z)}{\partial z} = -\exp(\log K_{d0} - \frac{E_d}{RT(z)}) \quad (14)$$

$$\frac{4v_f}{\rho D^2} \frac{\partial \log C(z)}{\partial z} = -\exp(\log K_{d0_nut} - \frac{E_{d_nut}}{RT(z)})$$

donde v_f es el caudal volumétrico, $\log K_{d0}$ y $\log K_{d0_nut}$ las constantes cinéticas de destrucción (1er orden) para las células y los nutrientes, E_{d_nut} y E_d las energías de activación para el microorganismo y el nutriente, R la constante universal de los gases, $\log N$ y $\log C$ la concentración de microorganismo y nutrientes, T la temperatura del líquido y D el diámetro de la conducción.

Balance de energía:

$$(1-x_i)v_f \cdot r C_p \frac{\partial T(z)}{\partial z} = Q(z) - \frac{3}{4} x_p \rho D^2 / Rd \cdot h(T(z) - T_s(z,1)) \quad (15)$$

donde ρ es la densidad del fluido, C_p su capacidad calorífica, h el coeficiente de transferencia de calor, Rd el radio de las partículas, T y T_s las temperaturas del fluido y de las partículas y Q el calor intercambiado.

SECCION MANTENIMIENTO (Partículas sólidas)

Balance materia:

$$\frac{4v_f}{\rho D^2} \frac{\partial \log N_s(z,r)}{\partial z} = -\exp(\log K_{d0} - \frac{E_d}{RT_s(z,r)}) \quad (16)$$

donde $\log N_s$ la concentración de microorganismo en las partículas.

Balance de energía:

$$\frac{4v_f}{\rho D^2} \frac{\partial T_s(z,r)}{\partial z} = \frac{k_s}{r_s C_{ps} R d^2} (\frac{\partial^2 T_s(z,r)}{\partial r^2} + \frac{2}{r} \frac{\partial T_s(z,r)}{\partial r})$$

$$\frac{\partial T_s(z,0)}{\partial r} = 0 \quad (17)$$

$$k_s \frac{\partial T_s(z,1)}{\partial r} = Rd \cdot h(T(z) - T_s(z,1))$$

donde T_s representa la temperatura para las partículas, C_{ps} la capacidad calorífica de las partículas, ρ_s su densidad, k_s su conductividad térmica y D es el diámetro de la conducción.

La implementación de este modelo (descrito por ecuaciones diferenciales parciales) en EcosimPro es parcialmente mostrada en un extracto correspondiente a los balances de materia y energía de las partículas sólidas en la sección de mantenimiento.

#include "\macros.h"

COMPONENT ProcAseptico

CONTINUOUS

--SECCION MANTENIMIENTO (Partículas sólidas)
-- Derivada 1ª respecto a z:

PDE_2D(0,Length,NLength,NRadius,TRUE,logNs,logNsz)

-- Derivada 1ª respecto a z:

PDE_2D(0,Length,NLength,NRadius,TRUE,Ts,Tsz)

```
-- Derivada 1ª respecto a r:
-- Condiciones Frontera:
EXPAND(i IN 1,NLength)
N2TsN[i] = ((h*Radius)/ks)*(T[i]-Ts[i,NRadius])
```

```
EXPAND(i IN 1,NLength)
N2Ts1[i] = 0.0
```

```
-- Derivada 2ª respecto a r:
PDE_2D_EXTR_2der(0,RadiusValue,NLength,
NRadius,FALSE,Ts,Tsr,Tsrr,TRUE,N2Ts1,TRUE,
N2TsN)
```

```
-- Balance de masa:
```

```
EXPAND(i IN 2,NLength)
EXPAND(j IN 1,NRadius)
```

```
logNs[i,j]'+(vf*4/pi/Diameter**2)*
logNsz[i,j]= -exp(logKdOEd/(R*Ts[i,j]))
```

```
-- Balance de Energia:
```

```
EXPAND(i IN 2,NLength)
Ts[i,1]'+(vf*4/pi/Diameter**2)*Tsz[i,1]=
(ks/(rhos*cps*Radius**2))*(3*Tsr[i,1])
```

```
EXPAND(i IN 2,NLength)
EXPAND(j IN 2,NRadius)
```

```
Ts[i,j]'+(vf*4/pi/Diameter**2)*Tsz[i,j]=
ks/(rhos*cps*Radius**2)*(Tsr[i,j]+
(2/((j-1)*DR))*Tsr[i,j])
```

8 MEJORAS CONSEGUIDAS CON LAS MACROS

La principal y más importante mejora consiste en la considerable reducción del tiempo de CPU necesario para simular los modelos cuando el integrador seleccionado es DASSL_SPARSE. En este caso se hace uso de la información sparse generada internamente por EcosimPro.

Sin embargo, para los otros integradores (DASSL, RK4) no se obtienen reducciones del tiempo de CPU debido a lo anteriormente expuesto.

La tabla 3 muestra estos resultados (en un PIII/450 MHz) para la esterilización de alimentos en un envase cilíndrico bidimensional (ejemplo tomado de Banga y co. [2]). Cabe destacar la notable reducción del tiempo de simulación (de 55.06 a 7.84 segundos) cuando hacemos uso de las macros y el integrador DASSL_SPARSE.

Tabla 3: Comparación de tiempos de CPU (en segundos) implementando el método NMOL como funciones de EcosimPro y como macros.

	DASSL SPARSE	DASSL	RK4
FUNC. NMOL	55.06	70.07	0.42
MACRO	7.84	69.58	0.43

9 CONCLUSIONES

En este trabajo se ha presentado la implementación mediante macros del método de las líneas (NMOL). Este procedimiento permite declarar en EcosimPro, de un modo sencillo, sistemas descritos por ecuaciones diferenciales parciales.

Esta herramienta puede ser enormemente eficaz y útil cuando queramos implementar y simular procesos reales descritos por varias ecuaciones diferenciales en derivadas parciales, que pueden implicar, tras la discretización, miles de DAEs. En estos casos, la reducción del tiempo de CPU necesario para ejecutar cada simulación será un factor crítico, y hemos demostrado cómo afrontarlo con éxito mediante el uso de integradores dispersos.

Agradecimientos

Agradecemos a Pedro Cobas la ayuda prestada en el desarrollo de este trabajo.

Referencias

- [1] Armenante, P., Leskowitz, M., (1990) Design of continuous sterilization systems for fermentation media containing suspended solids, *Biotechnology Progress*, 6, pp. 292-306.
- [2] Banga, J.R., Martín, R.P., Gallardo, J.M. and Casares, J.J., (1991) "Optimization of thermal processing of conduction-heated canned foods: Study of several objective functions", *J. Food Engineering*, 14(1), pp. 25-51.
- [3] Li, S., (1998) Adaptive Mesh Methods and Software for Time-Dependent Partial Differential Equations, PhD thesis, Univ. of Minnesota, USA.
- [4] Li, S. and Petzold, L. (1999) Design of new {DASPK} for sensitivity analysis, Technical report, UCSB.
- [5] Oh, M. (1995) Modelling and Simulation of Combined Lumped and Distributed Processes, PhD thesis, Department of Chemical Engineering and Chemical Technology, Imperial College of Science, Technology and Medicine.

- [6] Schiesser, W.E., (1991) The Numerical Method of Lines, Academic Press, New York.
- [7] Schiesser, W.E., (1994) Computational Mathematics in Engineering and Applied Science: ODEs, DAEs and PDEs, CRC Press, Inc.
- [8] Silebi, C. and Schiesser, W. (1992) Dynamic Modeling of Transport Process Systems, Academic Press, London.