**Departamento de Matemática Aplicada a la Ingeniería Industrial**

**Guion para la Práctica 2 de Ecuaciones Diferenciales (grupos de mañana)**

**Una introducción al estudio de EDPs de primer orden mediante MatLab**

---

# Índice


**1. Introducción** **1**

**2. Representación gráfica con MatLab** **1**

2.1. Ejemplo en MatLab . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1

2.2. Curvas características . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

2.3. Representación de la solución . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3


---

# 1. Introducción

El objetivo de esta práctica es utilizar la herramienta MatLab para estudiar la solución de una Ecuación en Derivadas Parciales de primer orden, con condición inicial, del tipo:

$$\begin{cases} u_t + c\left(x,u\right) u_x = 0, & \text{en } \mathbb{R}\times(0,\infty)\,, \\ u\left(x,0\right) = u_0\left(x\right), & x\in\mathbb{R}, \end{cases}$$

donde la función $c$ puede depende de $x$ y/o $u$. En general este tipo de EDPs se resuelve calculando las curvas características asociadas a la EDP. Puesto que el lado derecho de la EDP es nulo, en este caso concreto la solución $u$ será constante en dichas curvas y por tanto, la solución puede representarse gráficamente sabiendo como se desplazan los cada punto inicial $x_0$ a lo largo de estas curvas incluso aunque no pueda calcularse explícitamente la solución.

# 2. Representación gráfica con MatLab

En este tipo de EDPs es especialmente importante la representación gráfica por un lado de las curvas características y por otro lado de la solución $u$.

## 2.1. Ejemplo en MatLab

Vamos a considerar el siguiente problema de valor inicial:

$$\begin{cases} u_t - x u_x = 0, & \text{en } \mathbb{R}\times(0,\infty)\,, \\ u\left(x,0\right) = e^{-(x-1)^2}, & x\in\mathbb{R}. \end{cases}$$

Para calcular la solución de este problema primero fijamos un valor $x_0 \in \mathbb{R}$ y calculamos la curva característica que verifica que $x\left(0\right) = x_0$. Tras realizar los cálculos, puede comprobarse que en este caso

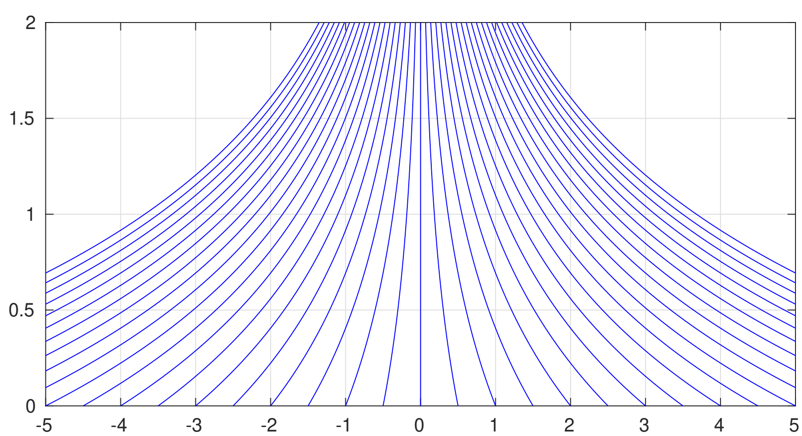
$$x\left(t\right) = x_0 e^{-t}.$$

## 2.2. Curvas características

La representación gráfica de las curvas características en función de cada $x_0 \in \mathbb{R}$ es muy importante en general. Para realizar esta representación gráfica en MatLab, abriremos un nuevo *script* y haremos lo siguiente:

- **Paso 1. Definimos un vector $t$ con la partición del intervalo temporal que queremos representar:**
```
>> t=0:0.01:2;
```
- **Paso 2. Hacemos un pequeño bucle recorriendo distintos valores de $x_0$ y dibujamos una curva característica para cada $x_0$:**
```
>> for x0=-10:0.5:10
>>     xc=x0*exp(-t);    % curva caracteristica para cada valor x0 (en MatLab x0)
>>     plot(xc,t,'b');    % dibujo de la curva característica (para dibujar todas
                            las curvas en azul utilizamos 'b')
>>     hold on    % para que al dibujar la siguiente curva no borre las anteriores
>> end
```
- **Paso 3. Añadimos algunos comando para visualizar mejor la gráfica:**
```
>> hold off    % para no seguir dibujando sobre la misma gráfica
>> grid    % para visualizar una malla
>> axis([-5 5 0 2])    % para ajustar los ejes
```

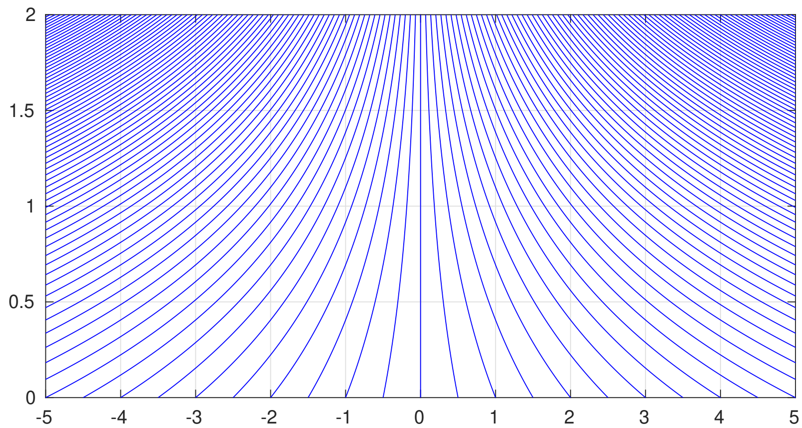
A partir de aquí podemos realizar un primer análisis de la solución, por ejemplo, si las curvas características rellenan todo el semiplano o existen regiones que no pertenezcan a alguna curva característica, si las curvas características cortan entre sí y se produce alguna onda de choque, etc.




En un primer vistazo vemos como las curvas características no se cortan entre sí. Aunque además parece que las curvas no rellenan todo el plano esto se debe al rango de valores $x_0$ que hemos utilizado en el bucle. Modificando de la siguiente manera:

- `>> for x0=-50:0.5:50`

ya podemos observar como efectivamente las curvas características rellenan todo el semiplano $t > 0$:




## 2.3. Representación de la solución

La representación de la solución es relativamente sencilla en MatLab una vez tenemos el cálculo de las curva características. La idea sería representar la solución en distintos instantes de tiempo para ver como evoluciona. Nuevamente abriremos un nuevo script y seguimos los siguiente pasos:

- **Paso 1. Definimos la condición inicial, y el vector x0.** Este último lo utilizaremos para representar la solución y mover a lo largo de las curvas características:
```
>> u0=@(x) exp(-(x-1).^2);
>> x0=-5:0.01:5;
```

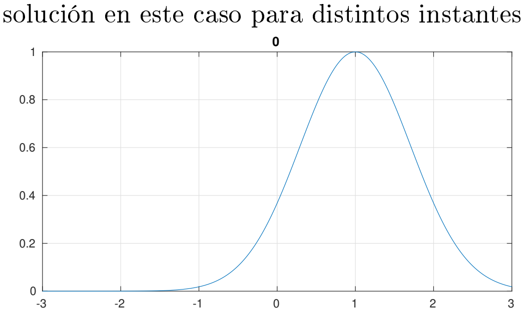
- **Paso 2. Hacemos un pequeño bucle recorriendo los distintos instantes de tiempo en los que queremos dibujar la solución:**
```
>> for t=0:0.1:4
>>      xc = x0*exp(-t);     % movemos los valores del vector x0 a lo largo de las
                               curvas características
>>      plot(xc,u0(x0))
>>      title(t)     % para que aparezca el instante de tiempo en la figura
>>      grid
>>      axis([-5 5 0 1])
>>      pause    % para la ejecución hasta que pulsamos una tecla
>> end
```

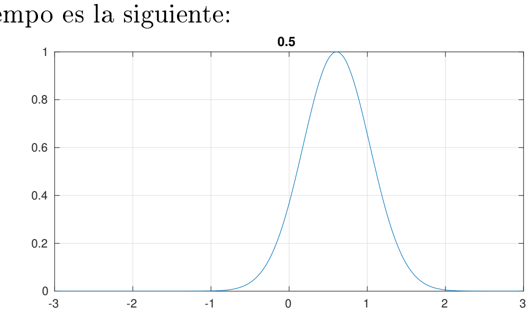
Observemos que si quisiéramos dibujar sólo la condición inicial hubiéramos utilizado el comando:

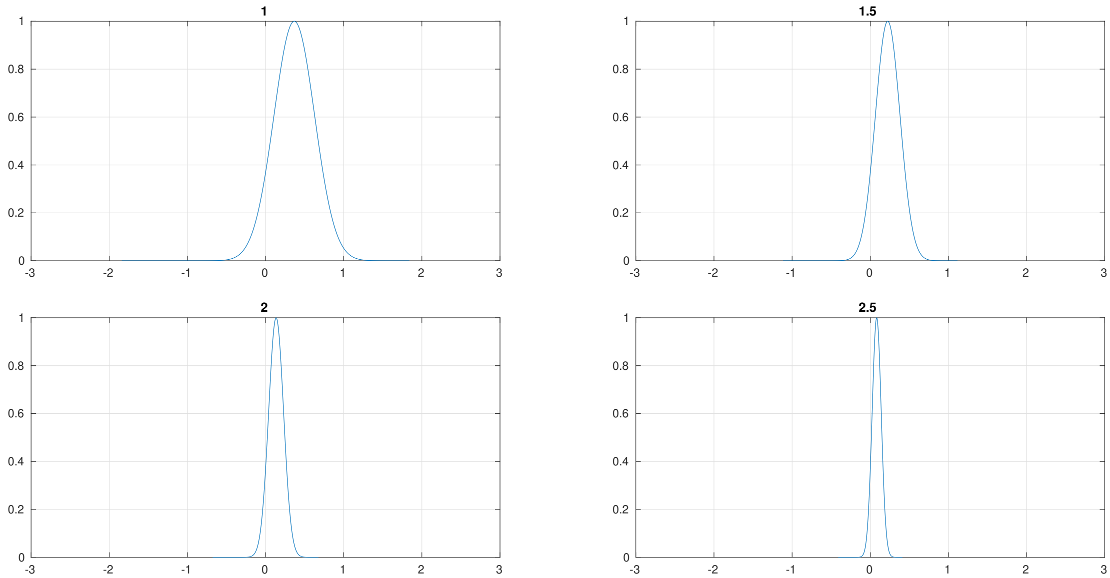
>> plot(x0,u0(x0))

y que por tanto lo único que hacemos dentro del bucle es cambiar x0 por xc, es decir, mover los puntos del vector x0 a través de las curvas características.

La solución en este caso para distintos instantes de tiempo es la siguiente:






donde podemos observar que no se produce ninguna onda de choque porque el resultado siempre es una función, es decir, a cada valor de $x$ un único valor de $u$ (que ya sabíamos por la gráfica de las curvas características).