Ecuaciones Diferenciales

1. Ecuaciones diferenciales: preliminares

- Definición
- Tipos: EDOs, EDPs, …

2. Métodos numéricos para ED ordinarias

Definición: Una ecuación diferencial es una ecuación que incorpora una (o varias variables) y sus derivadas.

$$U'(x) = U(x) \rightarrow \text{Solución: } U(x) = e^{x} + K$$

$$U''(x) = -U(x) \rightarrow \text{Solución: } U(x) = \left\{ \begin{array}{l} \cos(x) \\ \operatorname{sen}(x) \end{array} \right.$$

Si las variables dependen únicamente de una variable independiente (por ejemplo, el tiempo) se denominan **ecuaciones diferenciales ordinarias EDOs**, y las derivadas que aparecen son derivadas totales.

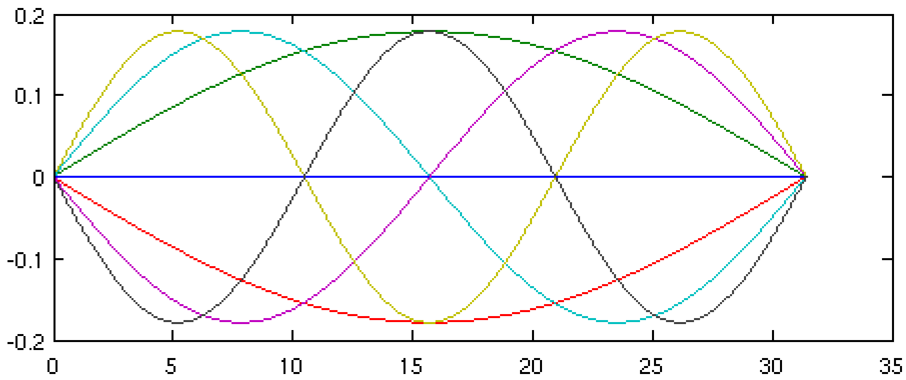
$$\frac{\partial^{2} U(x,t)}{\partial t^{2}} = c^{2} \frac{\partial^{2} U(x,t)}{\partial x^{2}}$$

Si por el contrario las variables dependen de más variables independientes (por ejemplo, espacio y tiempo a la vez), las ecuaciones diferenciales se denominan **en derivadas parciales EDPs**, y las derivadas que aparecen son derivadas parciales.

Una ecuación diferencial describe cómo varía una magnitud en el tiempo (o en el espacio) una variable. Es el concepto matemático más usado en física.
La solución de una ecuación es una **función** *U(t)* (o, mejor dicho, una familia de soluciones).

Vamos a trabajar unicamente con EDOs para resolver numéricamente dos tipos de problemas:

1.- Problema de contorno. (Boundary value problem)



$$\begin{cases} U''(x) + U(x) = 0 & \text{Ecuación diferencial} \\ U(0) = 1; U(p/2) = 0 & \text{Valor en los contornos} \end{cases}$$

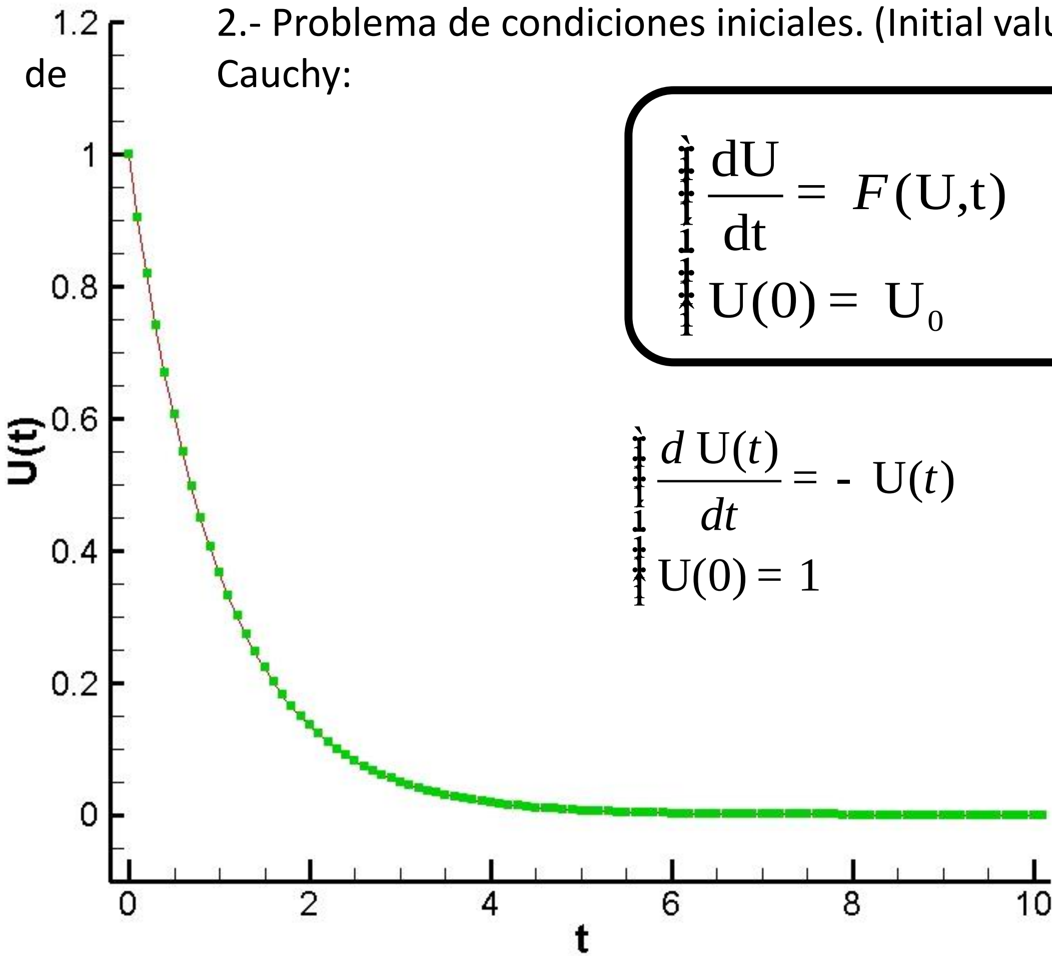
Una ecuación diferencial describe cómo varía una magnitud en el tiempo (o en el espacio) una variable. Es el concepto matemático más usado en física.
La solución de una ecuación es una **función** *U(t)* (o, mejor dicho, una familia de soluciones).

Vamos a trabajar unicamente con EDOs para resolver numéricamente dos tipos de problemas:

1.- Problema de contorno. (Boundary value problem)
2.- Problema de condiciones iniciales. (Initial value problem), tambien llamado problema de Cauchy:

$$\boxed{\begin{cases} \dfrac{dU}{dt} = F(U,t) \\ U(0) = U_0 \end{cases}}$$

$$\begin{cases} \dfrac{d\,U(t)}{dt} = -\,U(t) \\ U(0) = 1 \end{cases}$$



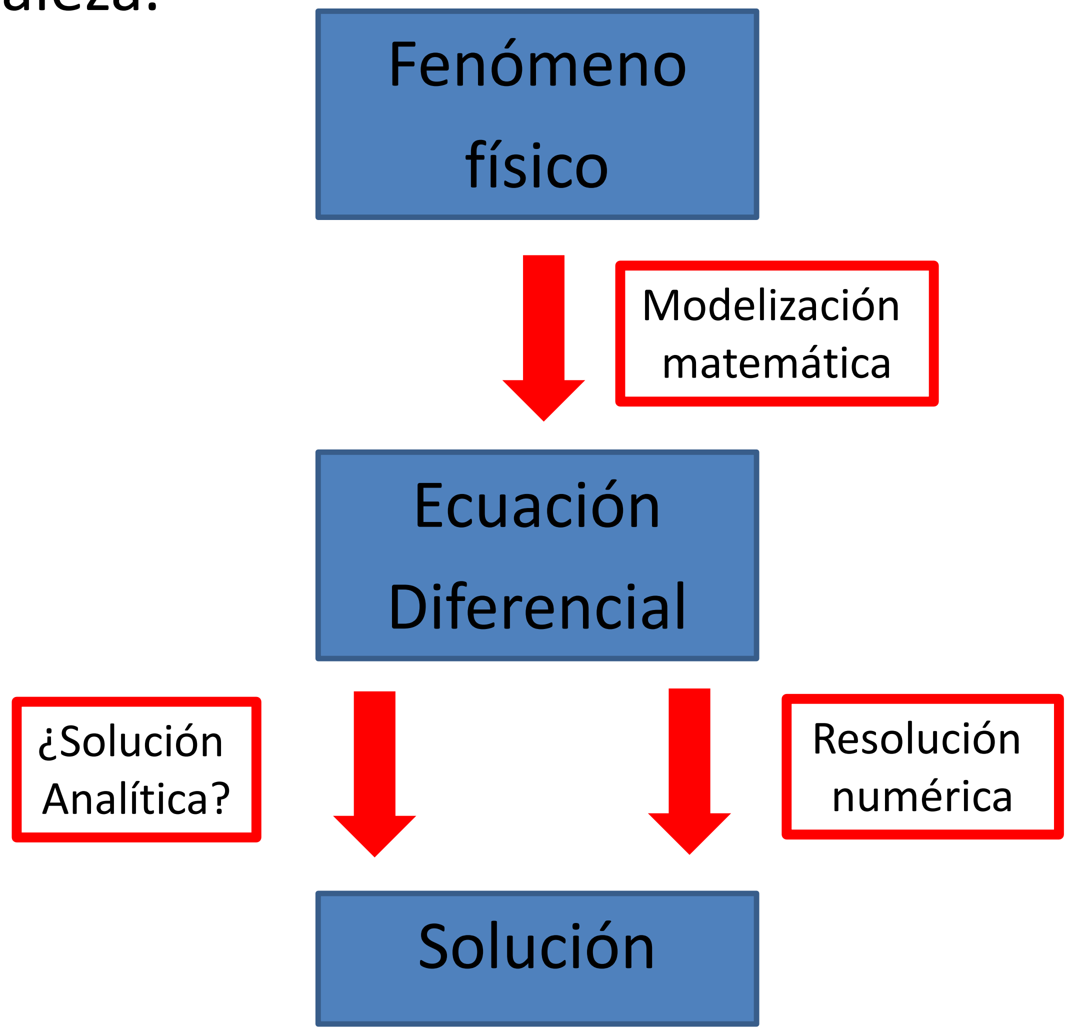
→ Desde el cálculo numérico, existen métodos para calcular la función solución. Si bien la solución es continua, la aproximación del método numérico será un conjunto de puntos que intenta interpolar a la función verdadera.

Ecuaciones Diferenciales

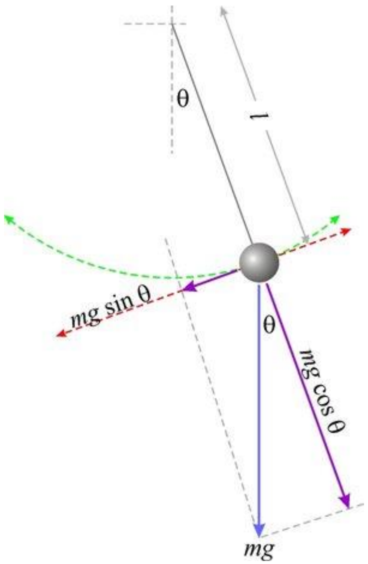
1. Ecuaciones diferenciales: preliminares

- Definición
- Tipos: EDOs
    - Ejemplos

- Aparecen como resultado de aplicar modelos matemáticos a problemas físicos observados en la naturaleza.

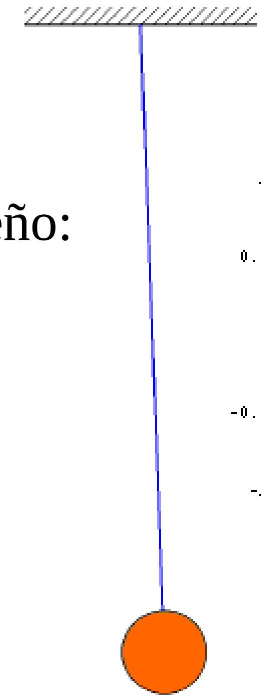


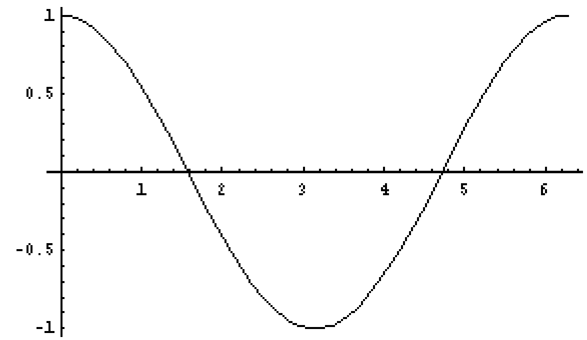
Movimiento del péndulo



2ª Ley de Newton:

$$\left.\begin{array}{l} F_t = m \cdot a_t = -\, mg\, sen\theta \\ a_t = l \cdot \dfrac{d^2\theta}{dt} = l \cdot \theta'' \end{array}\right\} \rightarrow \boxed{l \cdot \theta'' = -\, g\, sen\theta}$$





$\theta(t)$

Bajo la hipótesis de desplazamiento pequeño:

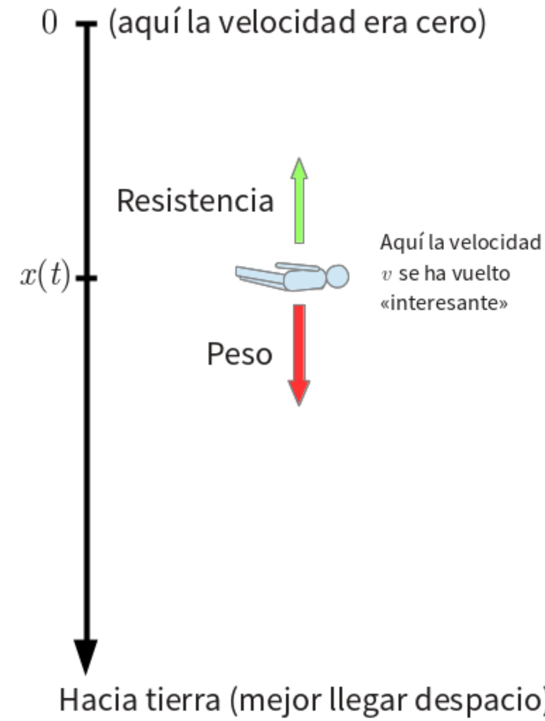
$$\mathrm{sen}\theta = \theta \rightarrow l \cdot \theta'' = -\, g\, \theta \rightarrow \boxed{\theta'' = -\, k^2 \theta}$$

$$\theta(t) = \begin{cases} c \cdot \cos(kt) \\ c \cdot \mathrm{sen}(kt) \end{cases}$$

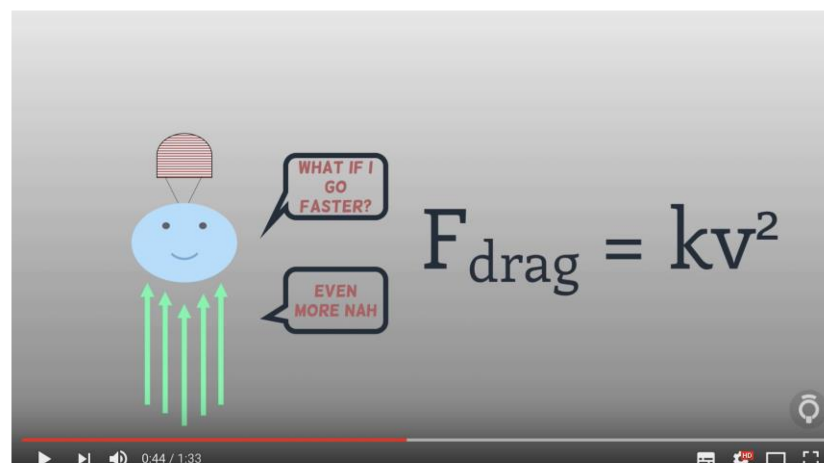
- Problema paracaidista (valor inicial)

$$F = ma \longrightarrow \frac{dv(t)}{dt} = F/m$$

$$\frac{dv(t)}{dt} = \frac{mg - Cv(t)^2}{m}$$



[elpinguinotolkiano]



Curvas planas

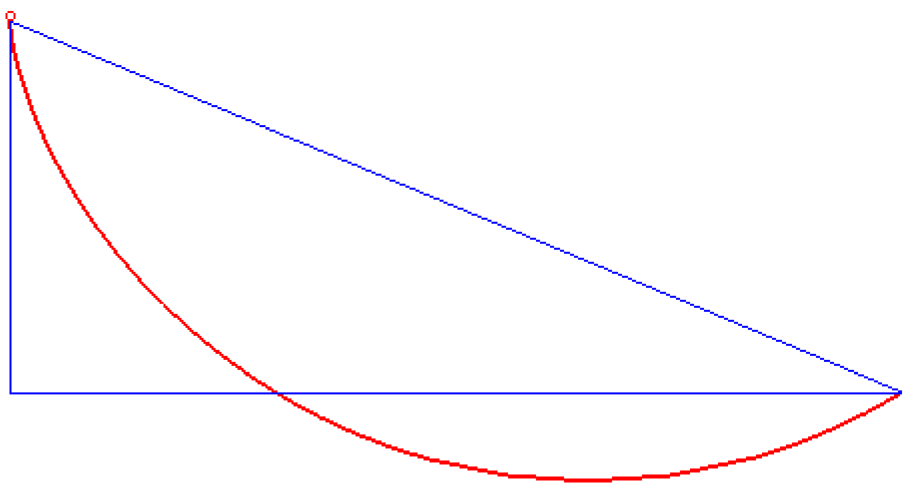
- Cicloide: Curva que describe un punto de una circunferencia al rodar esta sobre un plano.

$$\begin{cases} x = a\,(t - \sin t) \\ y = a\,(1 - \cos t) \end{cases}$$

Propiedades:

- Braquistocrona: (griego –menor tiempo)

¿Cuál es la curva más “rápida”?

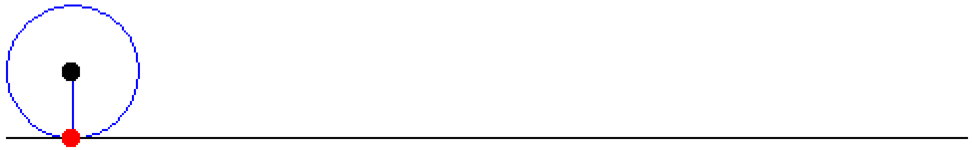


Link 

$$\boxed{\frac{dy}{dx} = \pm\sqrt{\frac{1}{2gC^2y} - 1}}$$

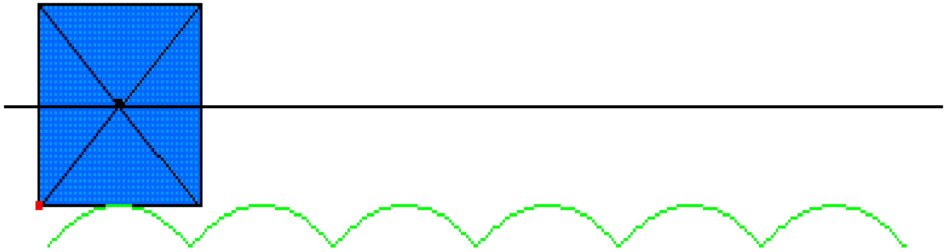
Curvas planas

- Cicloide: Curva que describe un punto de una circunferencia al rodar esta sobre un plano.



Propiedades:

- Tonterías varias

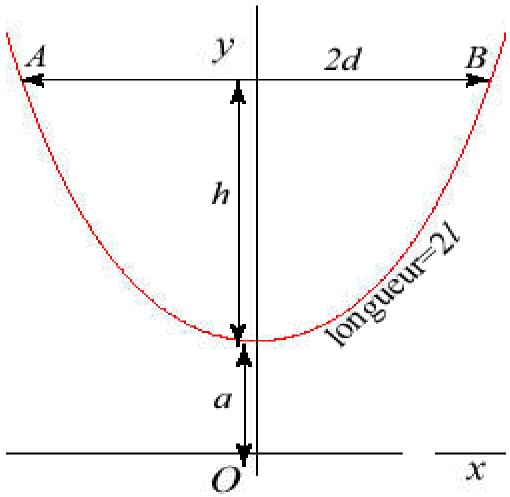


http://momath.org/

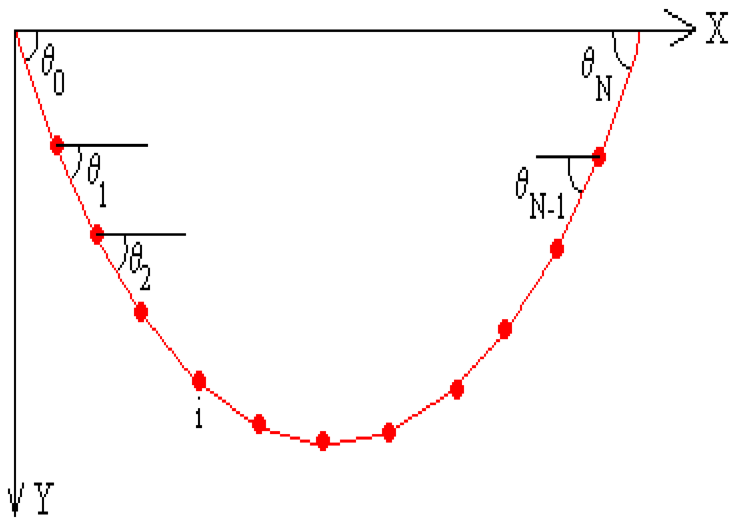
Curvas planas

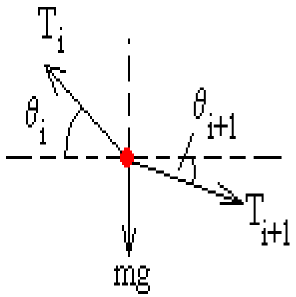
- Catenaria: curva de la cuerda colgante.

Ecuación: $y(x) = a\cosh\dfrac{x}{a}$



$$\boxed{\frac{d^2y}{dx^2} = \frac{\lambda}{T_H}\sqrt{1+\left(\frac{dy}{dx}\right)^2}}$$





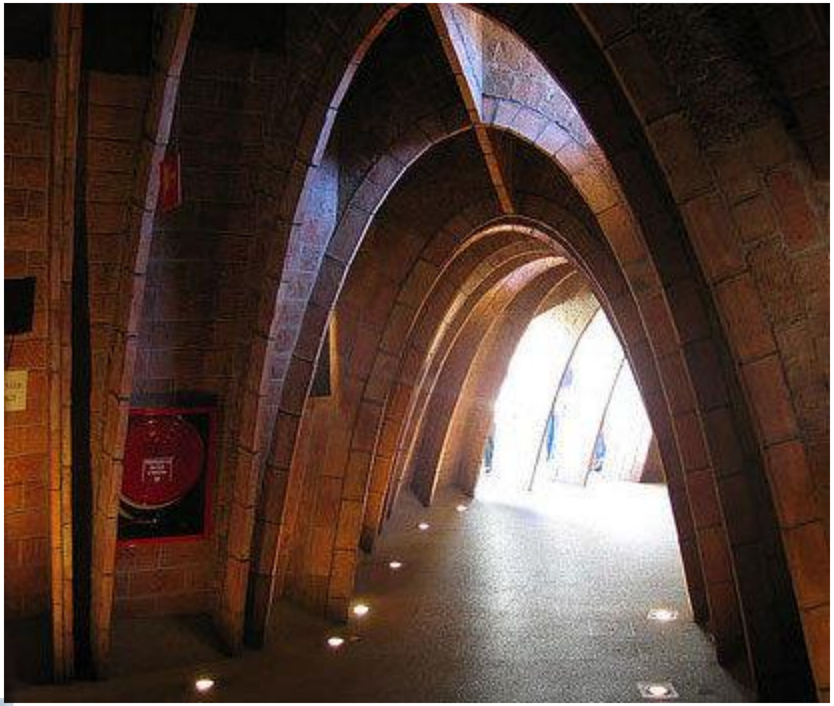
Galileo pensaba que la cuerda colgante era una parábola.

Huygens (1650) demostró a los 17 años que no era una parábola pero no fue capaz de deducir la ecuación verdadera.

Fue Bernuilli en 1691 quien, a partir de la descomposición de fuerzas (peso-tensión), encontró la ecuación.

Curvas planas

- Aplicaciones: Curva que no ejerce ninguna tensión transversal.



Dinámica de poblaciones

Ley de Malthus:

La poblacion crece exponencialmente:

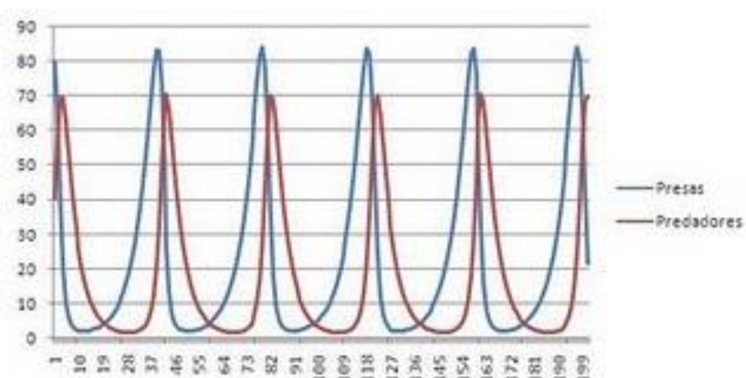
$$\frac{dP}{dt} = kP(t)$$

La cantidad de alimento crece linealmente:

$$\frac{dA}{dt} = cA_0$$

$\rightarrow$ Catástrofe Malthusiana

- Reproducción de bacterias
- # Operaciones computacionales al aumentar la complejidad
- Crecimiento de población en ausencia de depredadores
- Zombie Outbreak



Modelo de Lotka-Volterra

$$\begin{cases} \dfrac{dy}{dt} = y(a - bx) \\ \dfrac{dx}{dt} = -x(c - dx) \end{cases}$$

x = predadores; y = presas

Datación por radiocarbono

^{14}C **radiocarbono**, átomo de carbono con 14 neutrones en el núcleo

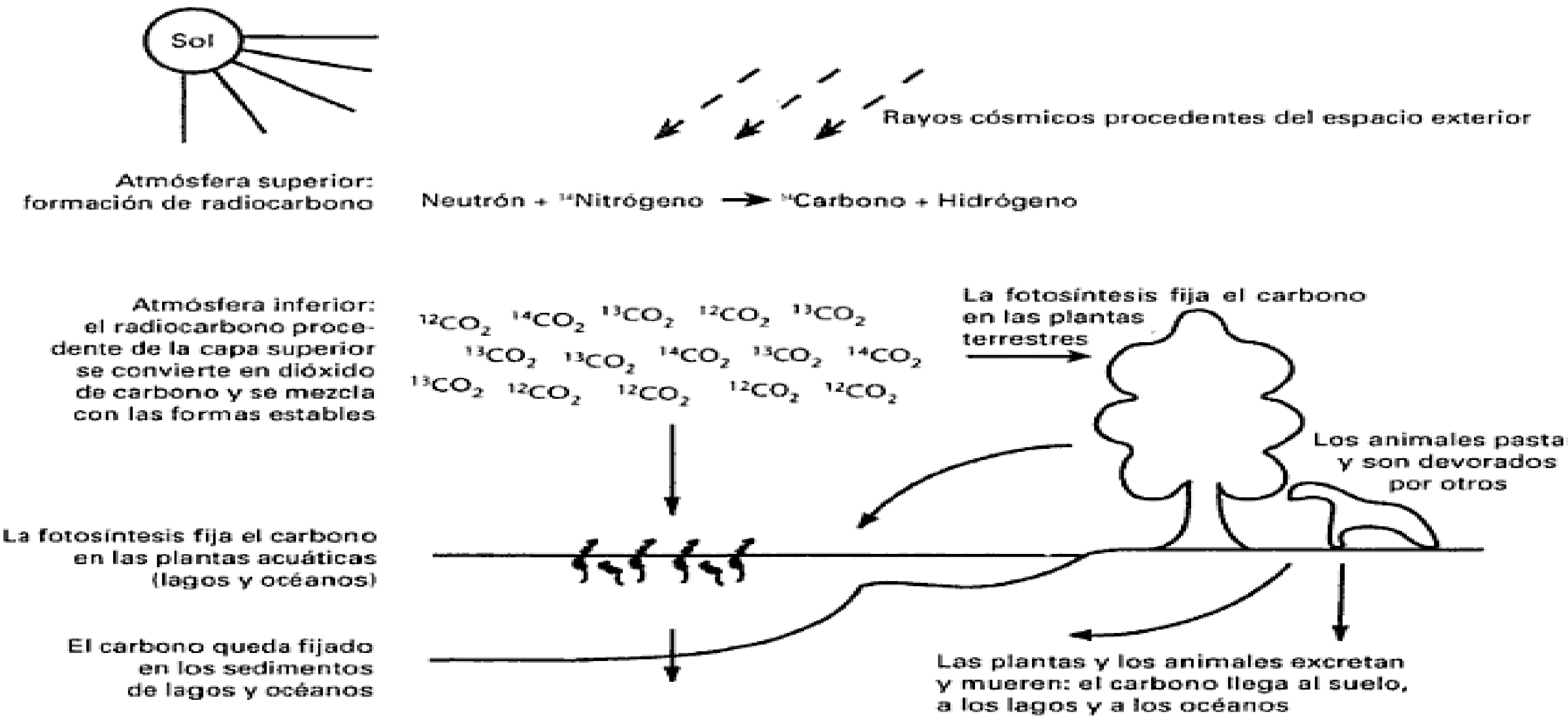


FIGURA 3.1. Formación y movimiento del radiocarbono en el medio ambiente.

Datación por radiocarbono

La variación de la masa (x') es función inversa de la tasa de desintegración radiactiva (T):

$$\frac{dx}{dt} = - T(x(t))$$

Como la tasa de desintegración sólo depende de la masa ($T = kx$)

$$\frac{dx}{dt} = - k \times x(t) \quad k \text{ constante de desintegración radiactiva.}$$

En el caso del ^{14}C el tiempo necesario para que la masa se reduzca a la mitad es de 5570 años.

William Libby.
Nobel de Química 1960
Datación por isótopos de ^{14}C

nature.com | about npg | news@nature.com | naturejobs | natureevents | help | site index

nature

my account | e-alerts | subscribe

SEARCH JOURNAL | Go

Thursday 07 Octob

Journal Home
Current Issue
AOP
Archive

THIS ARTICLE
Download PDF
References
Export citation
Export references
Send to a friend
More articles like this
Table of Contents
< Previous | Next >

article

Nature 337, 611 - 615 (16 February 1989); doi:10.1038/337611a0

Radiocarbon dating of the Shroud of Turin

P. E. DAMON[*], D. J. DONAHUE[†], B. H. GORE[*], A. L. HATHEWAY[†], A. J. T. JULL[*], T. W. LINICK[†], P. J. SERCEL[†], L. J. TOOLIN
R. E. M. HEDGES[‡], R. HOUSLEY[‡], I. A. LAW[‡], C. PERRY[‡], G. BONANI[§], S. TRUMBORE [||£], W. WOELFLI[§], J. C. AMBERS[¶], S
N. LEESE[¶] & M. S. TITE[¶]

[*]Department of Geosciences, University of Arizona, Tucson, Arizona 85721, USA
[†]Department of Physics, University of Arizona, Tucson, Arizona 85721, USA
[‡]Research Laboratory for Archaeology and History of Art, University of Oxford, Oxford, OX1 3QJ, UK
[§]Institut für Mittelenergiephysik, ETH-H&0uml;nggerberg, CH-8093 Zürich, Switzerland
[||]Lament-Doherty Geological Observatory, Columbia University, Palisades, New York 10964, USA
[¶]Research Laboratory, British Museum, London, WC1B3DG, UK
[£]Present address: Institut für Mittelenergiephysik, ETH-Hönggerberg, CH-8093, Zurich, Switzerland.

Arizona 646 ± 31 a. p.
Oxford 750 ± 30 a. p.
Zurich 676 ± 24 a. p.
con intervalo de confianza del 95%

Ecuaciones Diferenciales

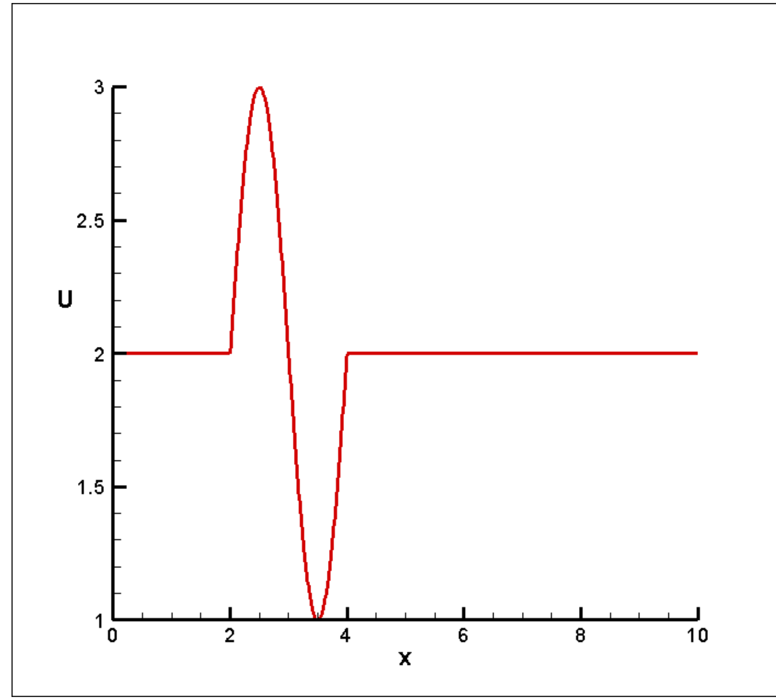
1. Ecuaciones diferenciales: preliminares

- Definición
- Tipos: EDPs
  - Ejemplos

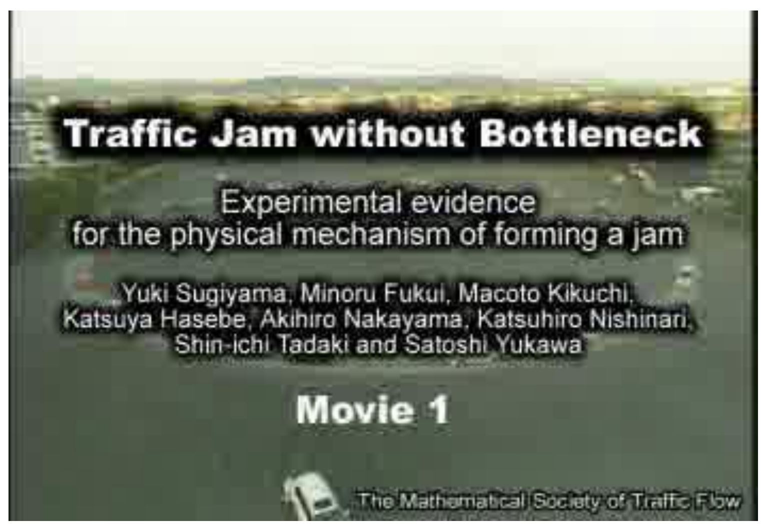
Ecuación de Burgers:

No viscosa $$\frac{\partial U(x,t)}{\partial t} + U(x,t)\frac{\partial U(x,t)}{\partial x} = 0$$

Viscosa $$\frac{\partial U(x,t)}{\partial t} + U(x,t)\frac{\partial U(x,t)}{\partial x} = \mu\frac{\partial^2 U(x,t)}{\partial x^2}$$

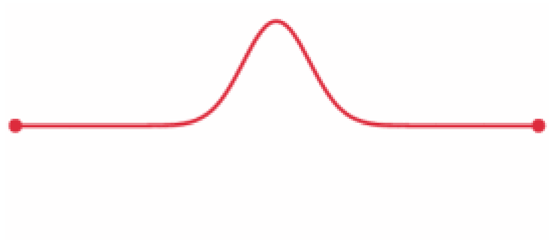


- Ondas de choque
- Modelos de tráfico



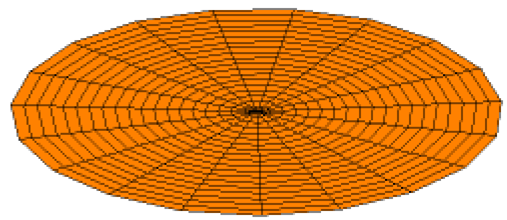
Ecuación de ondas:

$$\frac{\partial^2 U(\vec{x},t)}{\partial t^2} + c^2 \nabla^2 U(\vec{x},t) = 0$$



En una dimensión U(x,t):

$$\frac{\partial^2 U}{\partial t^2} = c^2 \frac{\partial^2 U}{\partial x^2}$$

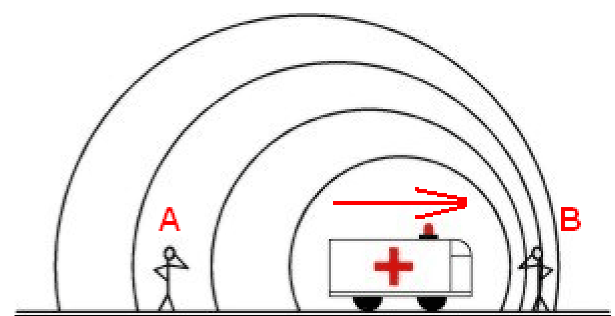


En dos dimensiones U(x,y,t):

$$\frac{\partial^2 U}{\partial t^2} = c^2 \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} \right)$$

En tres dimensiones U(x,y,z,t):

$$\frac{\partial^2 U}{\partial t^2} = c^2 \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} \right)$$



Ecuación de Navier-Stokes: (Aerodinámica-Mecánica de Fluidos)

$$\frac{\partial}{\partial t}\vec{U} + \nabla \cdot \vec{F} = 0$$

densidad

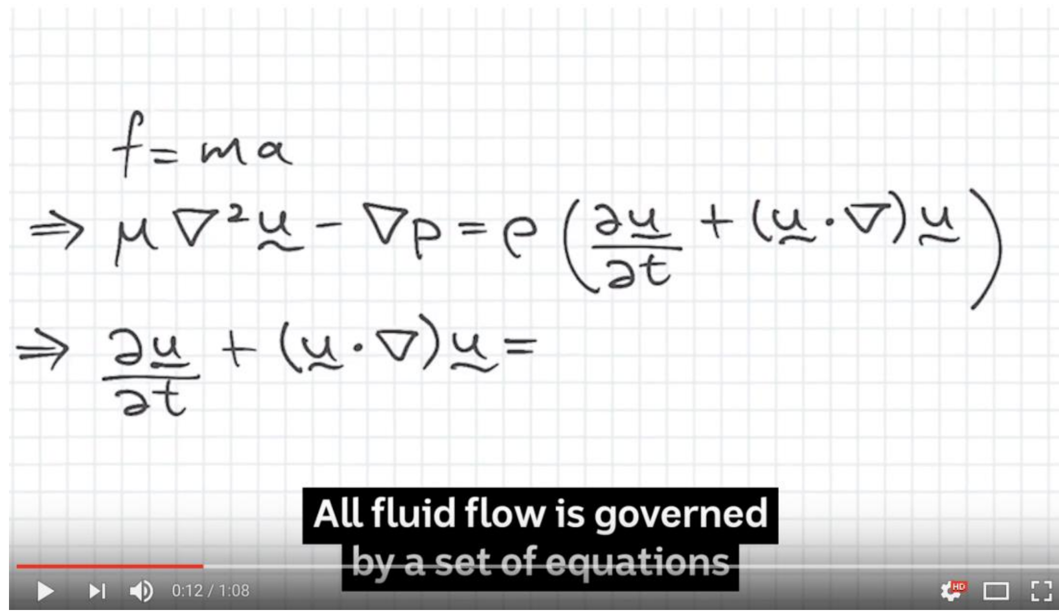
velocidad

energía

$$\vec{U} = \begin{vmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{vmatrix} \qquad \vec{F}(\vec{U}) = \begin{vmatrix} \rho u & \rho v & \rho w \\ \rho u^2 + P - \tau_{xx} & \rho u v - \tau_{xy} & \rho u w - \tau_{zx} \\ \rho u v - \tau_{xy} & \rho v^2 + P - \tau_{yy} & \rho w v - \tau_{zy} \\ \rho u w - \tau_{xz} & \rho v w - \tau_{yz} & \rho w^2 + P - \tau_{zz} \\ u(\rho E + P) - (\tau U)_x - q_x & v(\rho E + P) - (\tau U)_y - q_y & w(\rho E + P) - (\tau U)_z - q_z \end{vmatrix}$$

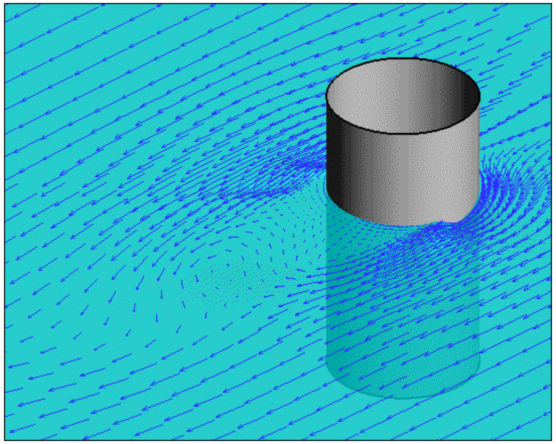
Problemas del milenio

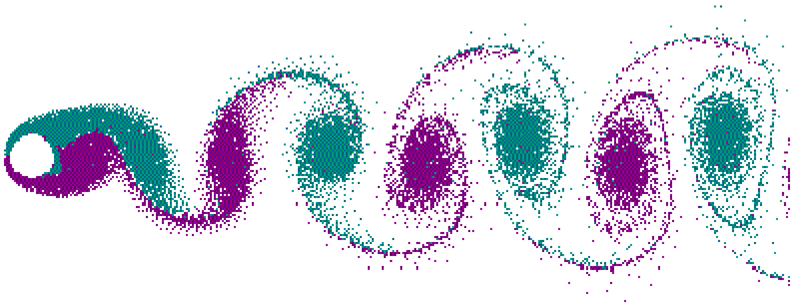
- P vs NP
- ~~Conjetura de Poincare~~
- Hipotesis de Riemann
- Navier- Stokes
- Conjetura de Hodge
- Yang-Mills y salto de masa
- Conjetura de Birch y Swinnerton-Dyer



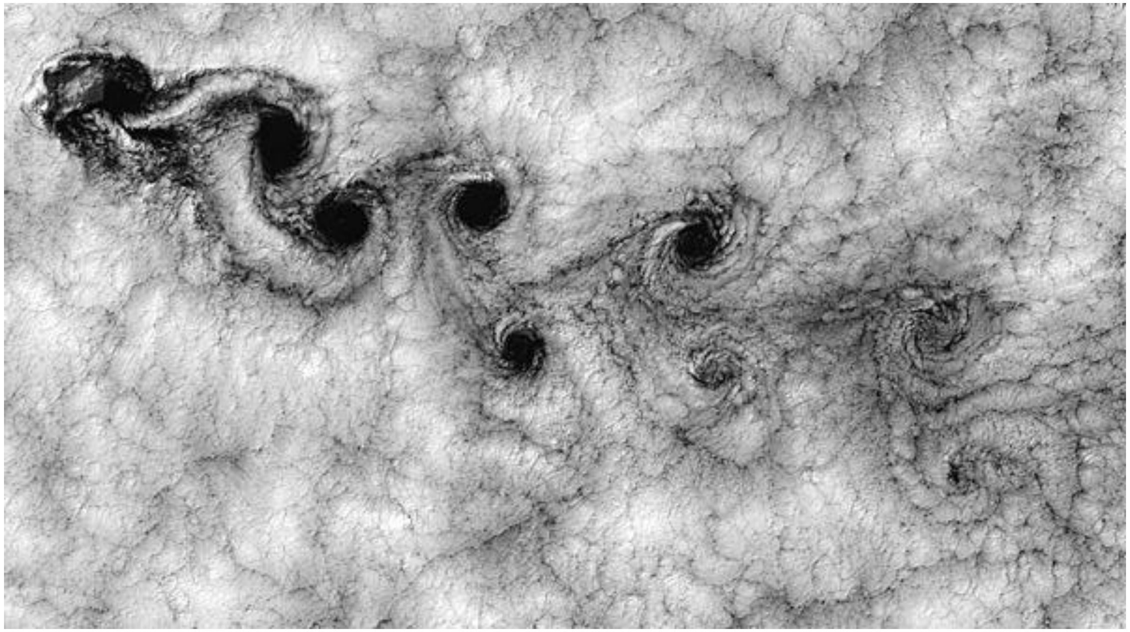
Ecuación de Navier-Stokes: (Aerodinámica-Mecánica de Fluidos)

Aplicación: Flujo alrededor de un cilindro





Calle de torbellinos de Karman



Islas de Juan Fernández (Chile)

Métodos numéricos para ED ordinarias: unipaso/multipaso

¿Cómo, a partir de una EDO, nos montamos un algoritmo cuya solución sea un conjunto de puntos que aproxima a la solución de la EDO?

Pasamos de una EDO (continua) a una ecuación en diferencias (discreta)

Bibliografía complementaria:

Cálculo Numérico en Ecuaciones Diferenciales Ordinarias, Juan A. Hernández, Aula Documental de Investigación

FORMULARIO DE DERIVADAS NUMÉRICAS -- RECORDATORIO

ADELANTADAS

n	$f^{(n)}(x)$
1	$f'(x) = \dfrac{f(x+h) - f(x)}{h} + O(h)$
2	$f''(x) = \dfrac{f(x+2h) - 2f(x+h) + f(x)}{h^2} + O(h)$
3	$f'''(x) = \dfrac{f(x+3h) - 3f(x+2h) + 3f(x+h) - f(x)}{h^3} + O(h)$
4	$f''''(x) = \dfrac{f(x+4h) - 4f(x+3h) + 6f(x+2h) - 4f(x+h) + f(x)}{h^4} + O(h)$

RETRASADAS

Da lugar a esquemas explícitos

n	$f^{(n)}(x)$
1	$f'(x) = \dfrac{f(x) - f(x-h)}{h} + O(h)$
2	$f''(x) = \dfrac{f(x) - 2f(x-h) + f(x-2h)}{h^2} + O(h)$
3	$f'''(x) = \dfrac{f(x) - 3f(x-h) + 3f(x-2h) - f(x-3h)}{h^3} + O(h)$
4	$f''''(x) = \dfrac{f(x) - 4f(x-h) + 6f(x-2h) - 4f(x-3h) + f(x-4h)}{h^4} + O(h)$

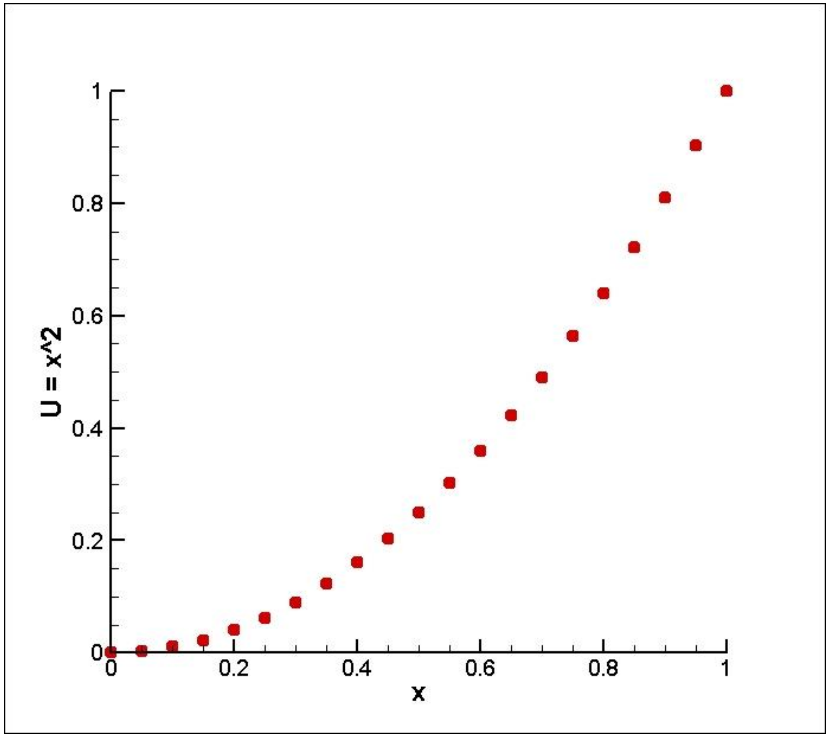
CENTRADAS

Da lugar a esquemas implícitos

n	$f^{(n)}(x)$
1	$f'(x) = \dfrac{f(x+h) - f(x-h)}{2h} + O(h^2)$
2	$f''(x) = \dfrac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2)$
3	$f'''(x) = \dfrac{f(x+2) - 2f(x+h) + 2f(x-h) - f(x-2h)}{2h^3} + O(h^2)$
4	$f''''(x) = \dfrac{f(x+2h) - 4f(x+h) + 6f(x) - 4f(x-h) + f(x-2h)}{h^4} + O(h^2)$

PROBLEMA DE CONTORNO (YA RESUELTO)

- ✓ Se basa en reescribir el problema diferencial en forma matricial y usar alguno de los algoritmos vistos para resolver problemas lineales.



$$x_i = \{0, 0.05, 0.1, 0.15 \ldots 0.95, 1\}$$

$$U(x_i) = x_i^2$$

Usando diferencias centradas:

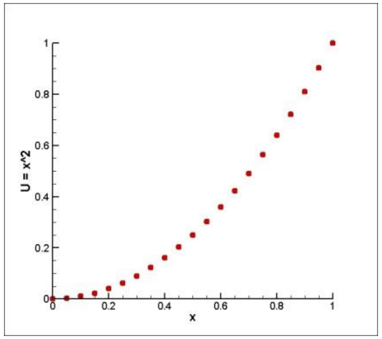
$$U'(x_i) = \frac{U(x_{i+1}) - U(x_{i-1})}{2 \cdot \Delta x}$$

Construimos la matriz de derivación

$$D\ U(x_i) = U'(x_i)$$

PROBLEMA DE CONTORNO (YA RESUELTO)

- ✓ Se basa en reescribir el problema diferencial en forma matricial y usar alguno de los algoritmos vistos para resolver problemas lineales.

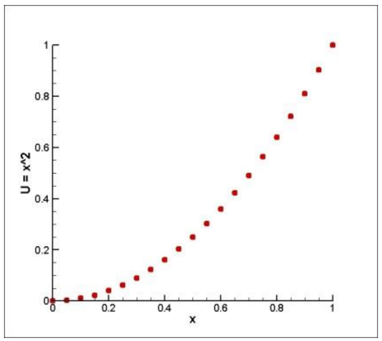


$$D\ U(x_i) = U'(x_i)$$

$$\begin{pmatrix} -\frac{1}{\Delta x} & \frac{1}{\Delta x} & & & & \\ \frac{-1}{2\Delta x} & 0 & \frac{1}{2\Delta x} & & & \\ & \frac{-1}{2\Delta x} & 0 & \frac{1}{2\Delta x} & & \\ & & \ddots & \ddots & \ddots & \\ & & & \frac{-1}{2\Delta x} & 0 & \frac{1}{2\Delta x} \\ & & & & -\frac{1}{\Delta x} & \frac{1}{\Delta x} \end{pmatrix} \begin{pmatrix} U(x_0) \\ U(x_1) \\ U(x_2) \\ \vdots \\ U(x_{n-1}) \\ U(x_n) \end{pmatrix} = \begin{pmatrix} U'(x_0) \\ U'(x_1) \\ U'(x_2) \\ \vdots \\ U'(x_{n-1}) \\ U'(x_n) \end{pmatrix}$$

PROBLEMA DE CONTORNO (YA RESUELTO)

- ✓ Se basa en reescribir el problema diferencial en forma matricial y usar alguno de los algoritmos vistos para resolver problemas lineales.



$$D\ U(x_i) = U'(x_i)$$

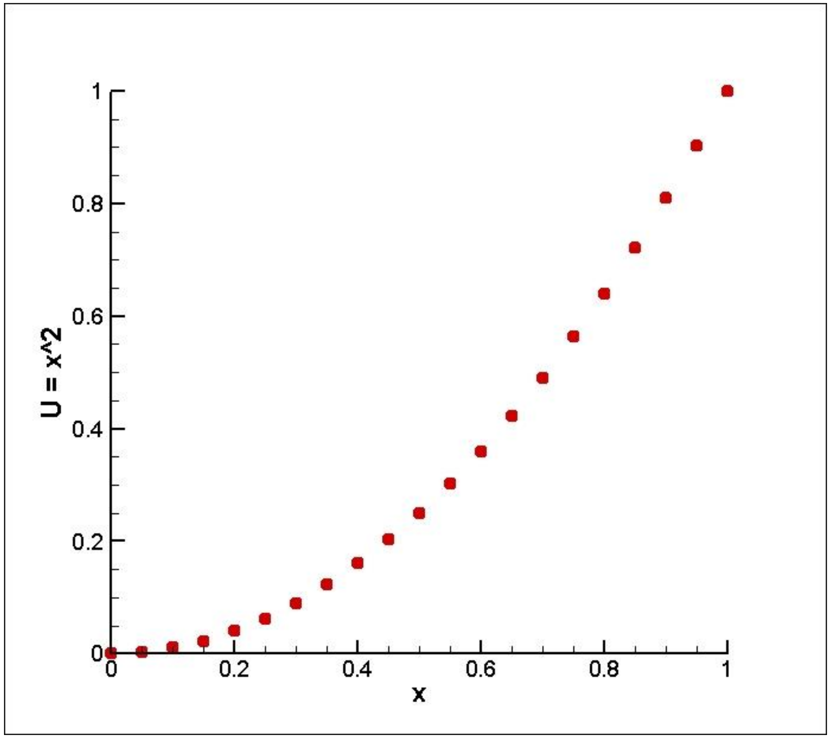
$$U'(x_1) = \frac{U(x_2) - U(x_0)}{2 \cdot \Delta x}$$

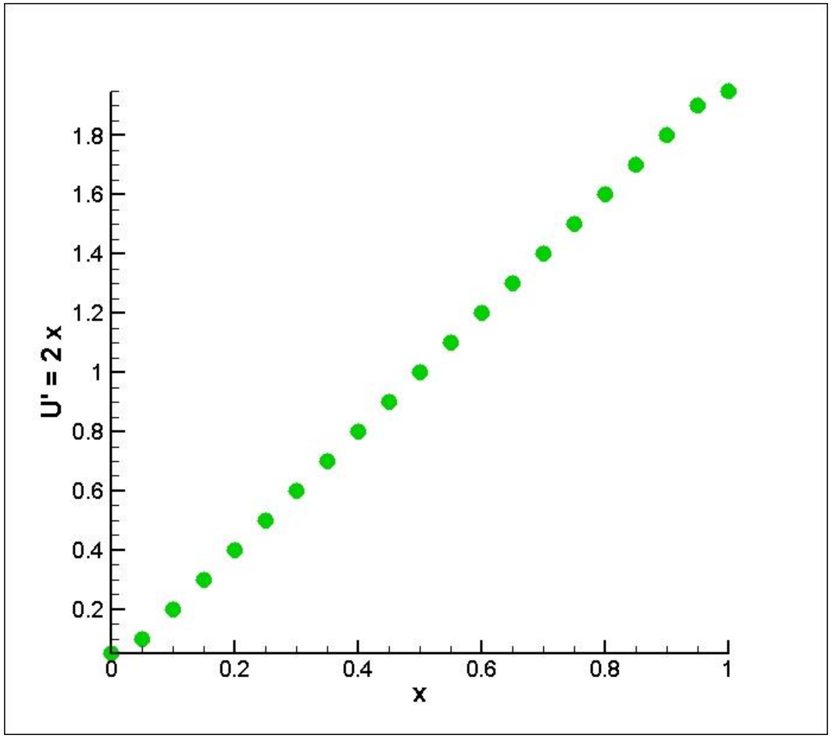
$$\begin{pmatrix} -\frac{1}{\Delta x} & \frac{1}{\Delta x} & & & & \\ \frac{-1}{2\Delta x} & 0 & \frac{1}{2\Delta x} & & & \\ & \frac{-1}{2\Delta x} & 0 & \frac{1}{2\Delta x} & & \\ & & \ddots & \ddots & \ddots & \\ & & & \frac{-1}{2\Delta x} & 0 & \frac{1}{2\Delta x} \\ & & & & -\frac{1}{\Delta x} & \frac{1}{\Delta x} \end{pmatrix} \begin{pmatrix} U(x_0) \\ U(x_1) \\ U(x_2) \\ \vdots \\ U(x_{n-1}) \\ U(x_n) \end{pmatrix} = \begin{pmatrix} U'(x_0) \\ \boxed{U'(x_1)} \\ U'(x_2) \\ \vdots \\ U'(x_{n-1}) \\ U'(x_n) \end{pmatrix}$$

PROBLEMA DE CONTORNO (YA RESUELTO)

- ✓ Se basa en reescribir el problema diferencial en forma matricial y usar alguno de los algoritmos vistos para resolver problemas lineales.

$$D\ U(x_i) = U'(x_i)$$

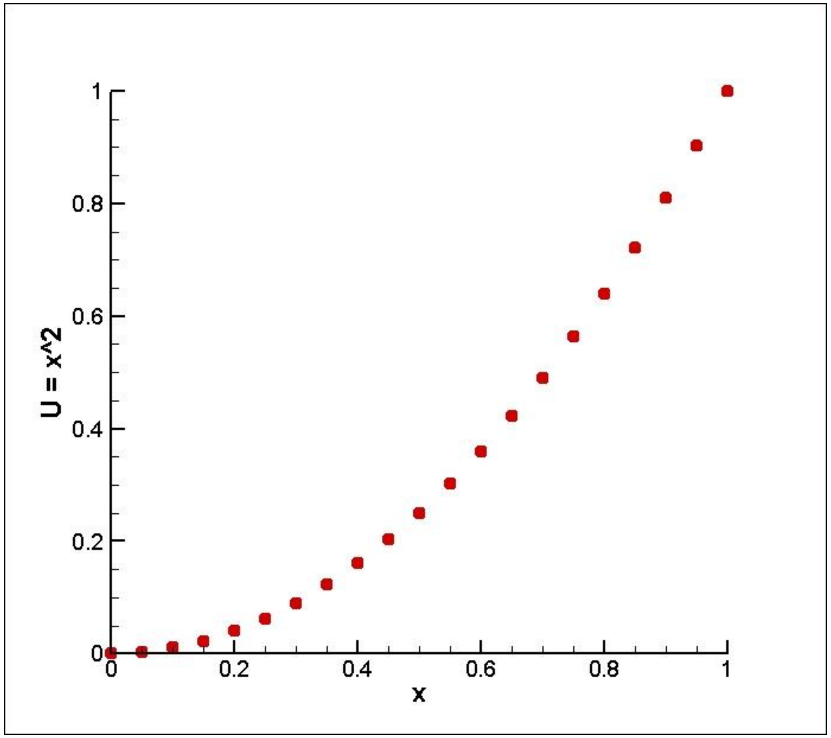


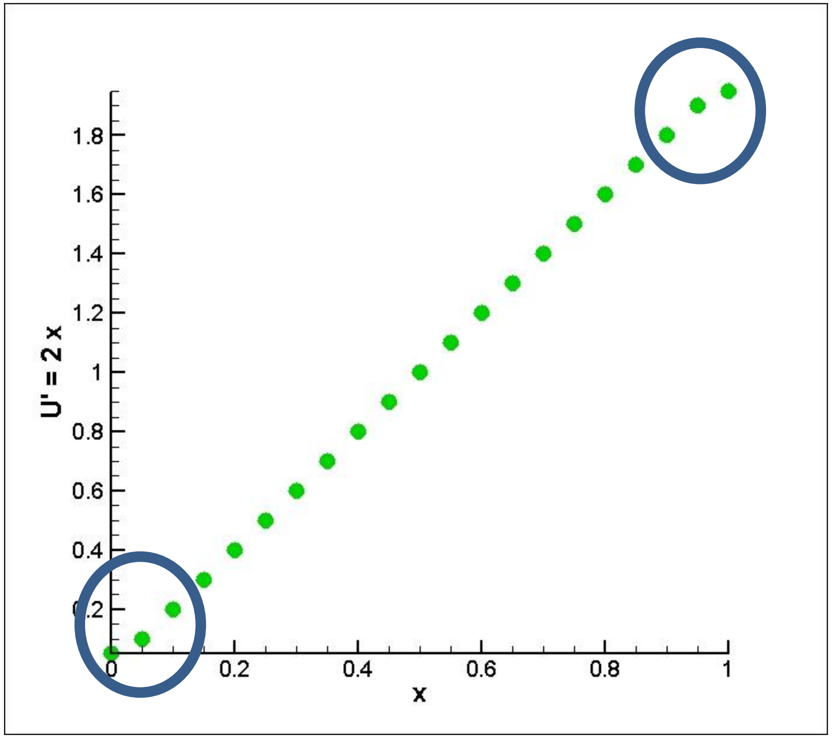


PROBLEMA DE CONTORNO (YA RESUELTO)

✓ Se basa en reescribir el problema diferencial en forma matricial y usar alguno de los algoritmos vistos para resolver problemas lineales.

$$D\ U(x_i) = U'(x_i)$$





PROBLEMA DE CONTORNO (YA RESUELTO)

- ✓ Se basa en reescribir el problema diferencial en forma matricial y usar alguno de los algoritmos vistos para resolver problemas lineales.

- ✓ Podemos usarlo para resolver el problema que usamos como ejemplo

$$\begin{cases} U''(x) + U(x) = 0 \\ U(0) = 1; U(p/2) = 0 \end{cases}$$

- ✓ Discretizamos el intervalo

$x_0 = 0$ $\qquad$ $x_n = \dfrac{p}{2}$

PROBLEMA DE CONTORNO (YA RESUELTO)

- ✓ Reescribiendo la ecuación diferencial en forma matricial

$$U''(x) + U(x) = 0 \;\Rightarrow\; M(U) = f(x)$$

- ✓ En este caso: $$M\big(U(x)\big) = 0 \text{ con } M = D \times D + I$$

siendo D la matriz de derivación anterior

$$M = \begin{pmatrix} O & & \\ & D & \\ & & O \end{pmatrix} \cdot \begin{pmatrix} O & & \\ & D & \\ & & O \end{pmatrix} + \begin{pmatrix} O & & \\ & 1 & \\ & & O \end{pmatrix}$$

Basta imponer los contornos:

PROBLEMA DE CONTORNO(YA RESUELTO)

Basta imponer los contornos: $\begin{cases} U''(x) + U(x) = 0 \\ U(0) = 1; U(\pi/2) = 0 \end{cases}$

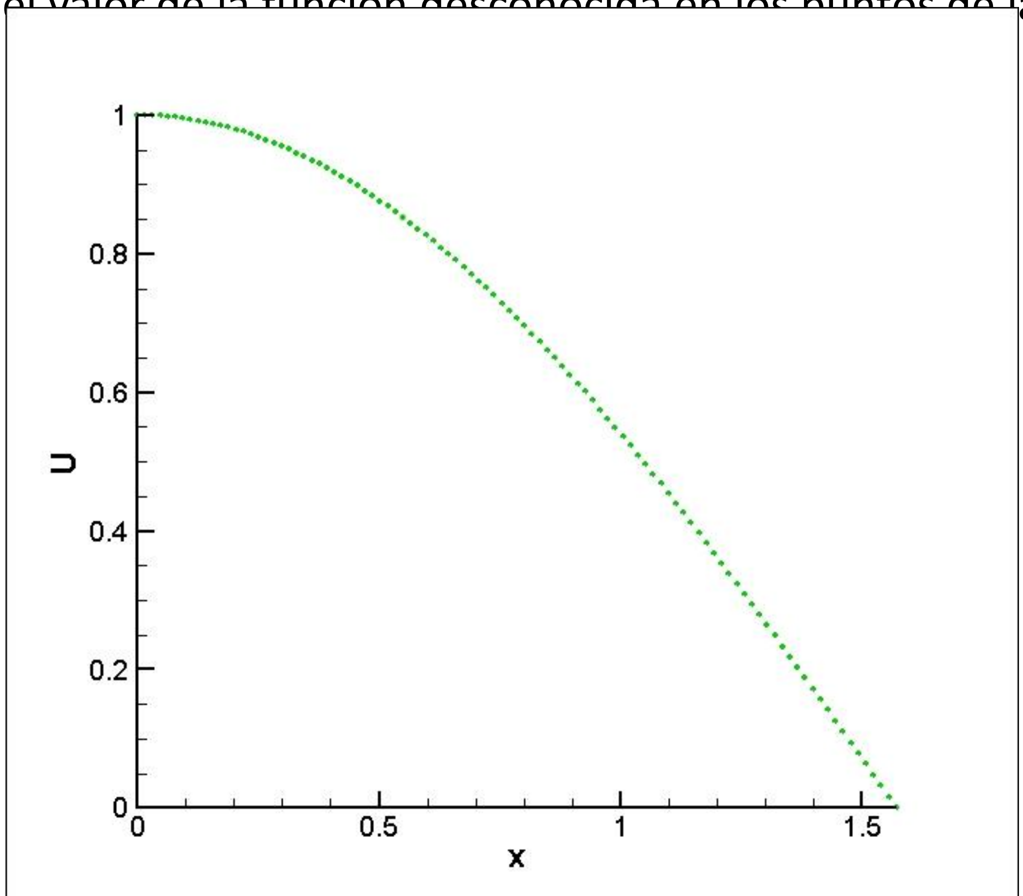
$$\begin{pmatrix} 1 & 0 & .. & . & 0 \\ M_{10} & . & . & . & M_{1n} \\ . & . & . & . & . \\ M_{n-10} & . & . & . & M_{n-1n} \\ 0 & . & . & . & 1 \end{pmatrix} \begin{pmatrix} U(x_0) \\ U(x_1) \\ . \\ . \\ U(x_n) \end{pmatrix} = \begin{pmatrix} 1 \\ F(x_1) \\ . \\ F(x_{n-1}) \\ 0 \end{pmatrix} \Rightarrow \begin{matrix} U(x_0) = U(0) = 1 \\ \\ U(x_n) = U(\pi/2) = 0 \end{matrix}$$

Queda un problema del tipo $Ax = b$ donde el vector de incógnitas está formado por el valor de la función desconocida en los puntos de la malla.

PROBLEMA DE CONTORNO (YA RESUELTO)

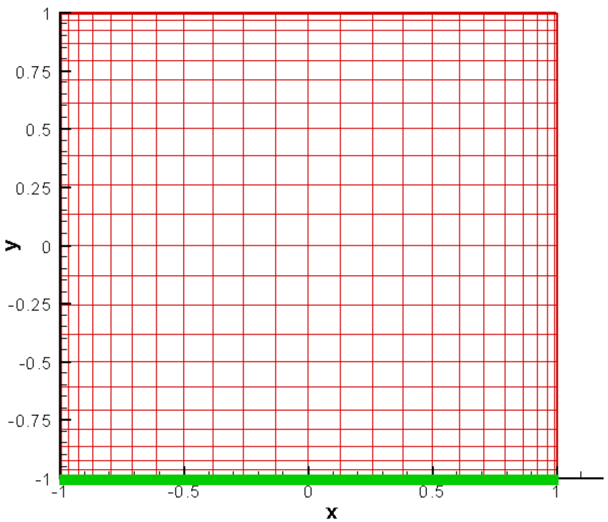
Basta imponer los contornos: $$\begin{cases} U''(x) + U(x) = 0 \\ U(0) = 1; U(\pi/2) = 0 \end{cases}$$

Queda un problema del tipo $Ax = b$ donde el vector de incógnitas está formado por el valor de la función desconocida en los puntos de la malla.



PROBLEMA DE CONTORNO: Extensión a 2D

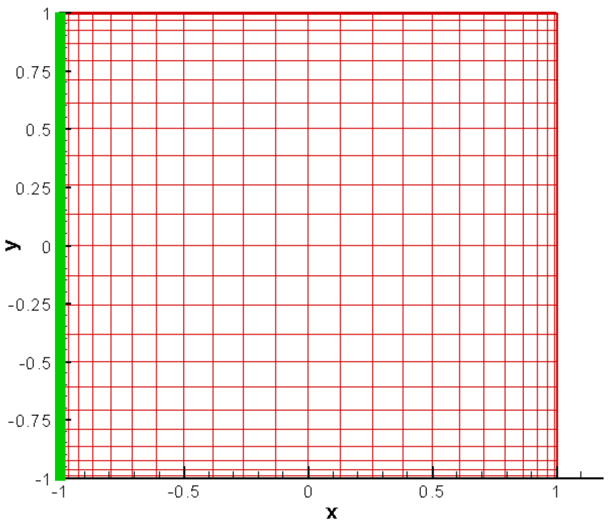
El dominio 2D se discretiza en forma de malla cartesiana que se transforma en un vector de variables



$$
\begin{pmatrix}
U_{0,0} \\
\cdot \\
\cdot \\
U_{N_x,0} \\
U_{0,1} \\
\cdot \\
\cdot \\
U_{N_x,1} \\
\\
\vdots \\
U_{N_x,N_y}
\end{pmatrix}
$$

PROBLEMA DE CONTORNO: Extensión a 2D

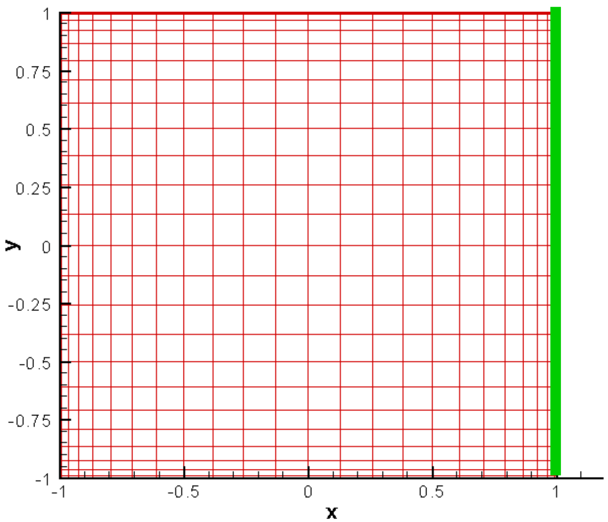
El dominio 2D se discretiza en forma de malla cartesiana que se transforma en un vector de variables



$$\Rightarrow \begin{pmatrix} \boxed{U_{0,0}} \\ \cdot \\ \cdot \\ U_{N_x,0} \\ \boxed{U_{0,1}} \\ \cdot \\ \cdot \\ U_{N_x,1} \\ \\ \\ \vdots \\ U_{N_x,N_y} \end{pmatrix}$$

PROBLEMA DE CONTORNO: Extensión a 2D

El dominio 2D se discretiza en forma de malla cartesiana que se transforma en un vector de variables



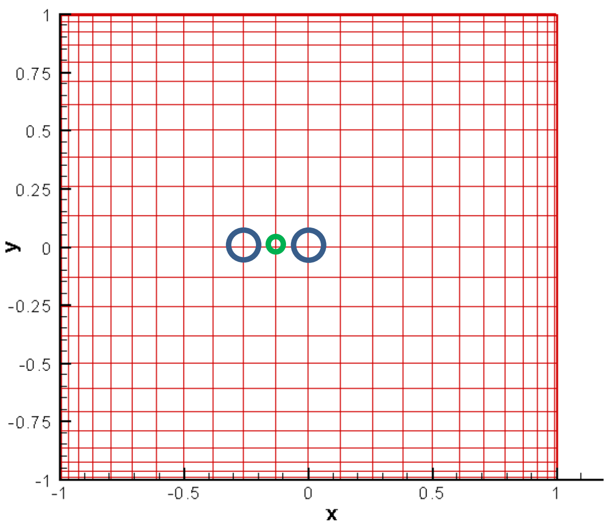
$$\Rightarrow \begin{pmatrix} U_{0,0} \\ \cdot \\ \cdot \\ \boxed{U_{N_x,0}} \\ U_{0,1} \\ \cdot \\ \cdot \\ \boxed{U_{N_x,1}} \\ \\ \\ \vdots \\ \boxed{U_{N_x,N_y}} \end{pmatrix}$$

PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

$$\frac{\partial U(x_i, y_j)}{\partial x} = \frac{U(x_{i+1}, y_j) - U(x_{i-1}, y_j)}{2 \cdot \Delta x}$$

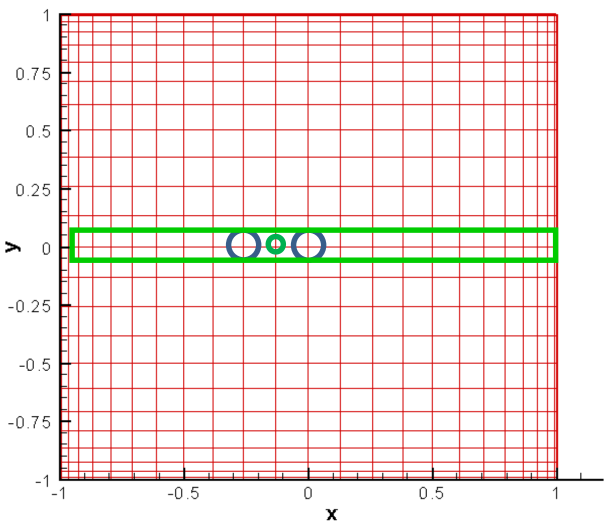
$$\begin{pmatrix} U(0,0) \\ U(1,0) \\ . \\ U(Nx,0) \\ ---- \\ \mathrm{M} \\ ---- \\ U(0,j) \\ \mathrm{M} \\ U(i-1,j) \\ U(i,j) \\ U(i+1,j) \\ \mathrm{M} \\ U(Nx,j) \\ ---- \\ \mathrm{M} \\ ---- \\ U(0,Ny) \\ U(1,Ny) \\ . \\ U(Nx,Ny) \end{pmatrix}$$



PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

$$\frac{\partial U(x_i, y_j)}{\partial x} = \frac{U(x_{i+1}, y_j) - U(x_{i-1}, y_j)}{2 \cdot \Delta x}$$



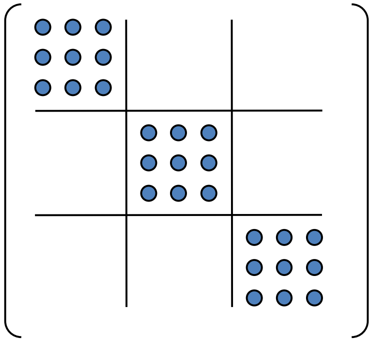
$$\begin{pmatrix} U(0,0) \\ U(1,0) \\ . \\ U(Nx,0) \\ ---- \\ \mathrm{M} \\ ---- \\ \boxed{\begin{matrix} U(0,j) \\ \mathrm{M} \\ U(i-1,j) \\ U(i,j) \\ U(i+1,j) \\ \mathrm{M} \\ U(Nx,j) \end{matrix}} \\ ---- \\ \mathrm{M} \\ ---- \\ U(0,Ny) \\ U(1,Ny) \\ . \\ U(Nx,Ny) \end{pmatrix}$$

PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

$$\frac{\partial U(x_i, y_j)}{\partial x} = \frac{U(x_{i+1}, y_j) - U(x_{i-1}, y_j)}{2 \cdot \Delta x}$$

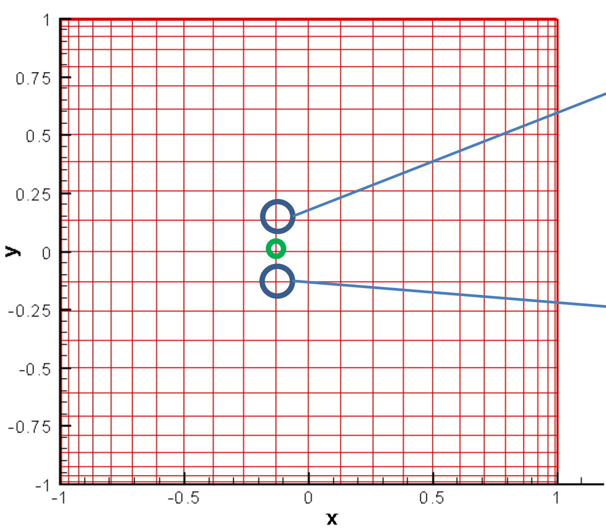
$$\left(\begin{array}{c|c|c|c}
\begin{matrix} d_{0,0} & d_{0,1} & \dots & d_{0,N} \\ d_{1,0} & d_{1,1} & \dots & d_{1,N} \\ & & \ddots & \\ d_{N,0} & d_{N,1} & \dots & d_{N,N} \end{matrix} & & & \\ \hline
 & \begin{matrix} d_{0,0} & d_{0,1} & \dots & d_{0,N} \\ d_{1,0} & d_{1,1} & \dots & d_{1,N} \\ & & \ddots & \\ d_{N,0} & d_{N,1} & \dots & d_{N,N} \end{matrix} & & \\ \hline
 & & \ddots & \\ \hline
 & & & \begin{matrix} d_{0,0} & d_{0,1} & \dots & d_{0,N} \\ d_{1,0} & d_{1,1} & \dots & d_{1,N} \\ & & \ddots & \\ d_{N,0} & d_{N,1} & \dots & d_{N,N} \end{matrix}
\end{array}\right)$$

$\mathcal{D}_x =$ 

PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

$$\frac{\partial U(x_i, y_j)}{\partial y} = \frac{U(x_i, y_{j+1}) - U(x_i, y_{j-1})}{2 \cdot \Delta y}$$

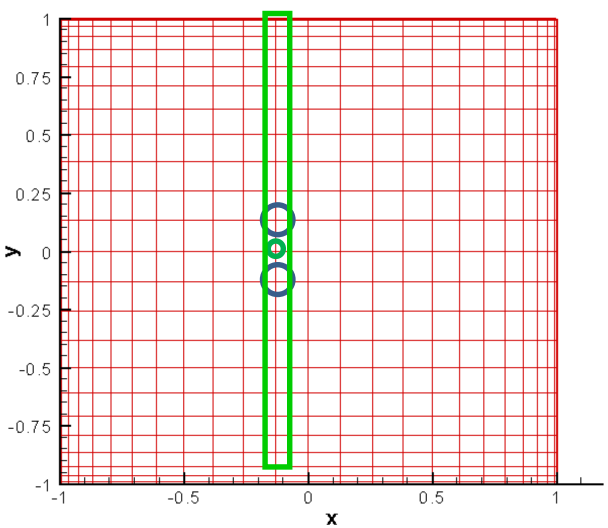


$$\begin{pmatrix} U(0,0) \\ U(1,0) \\ . \\ U(Nx,0) \\ ---- \\ \vdots \\ ---- \\ U(0, j-1) \\ \vdots \\ \boxed{U(i, j-1)} \\ \vdots \\ U(Nx, j-1) \\ ---- \\ U(0, j) \\ \vdots \\ \boxed{U(i, j)} \\ \vdots \\ U(Nx, j) \\ ---- \\ U(0, j+1) \\ \vdots \\ \boxed{U(i, j+1)} \\ \vdots \\ U(Nx, j+1) \\ ---- \\ \vdots \\ ---- \\ U(0, Ny) \end{pmatrix}$$

PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

$$\frac{\partial U(x_i, y_j)}{\partial y} = \frac{U(x_i, y_{j+1}) - U(x_i, y_{j-1})}{2 \cdot \Delta y}$$

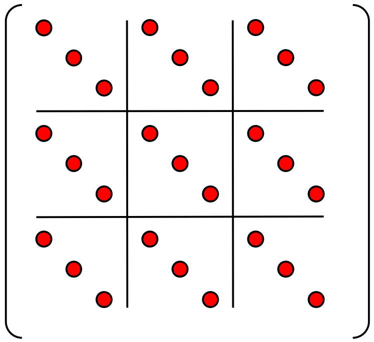


$$\begin{pmatrix} U(0,0) \\ U(1,0) \\ \boxed{\;.\;} \\ U(Nx,0) \\ ---- \\ \mathrm{M} \\ ---- \\ U(0,j-1) \\ \mathrm{M} \\ \boxed{U(i,j-1)} \\ \mathrm{M} \\ U(Nx,j-1) \\ ---- \\ U(0,j) \\ \mathrm{M} \\ \boxed{U(i,j)} \\ \mathrm{M} \\ U(Nx,j) \\ ---- \\ U(0,j+1) \\ \mathrm{M} \\ \boxed{U(i,j+1)} \\ \mathrm{M} \\ U(Nx,j+1) \\ ---- \\ \mathrm{M} \\ ---- \\ U(0,Ny) \end{pmatrix}$$

PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

$$\frac{\partial U(x_i, y_j)}{\partial y} = \frac{U(x_i, y_{j+1}) - U(x_i, y_{j-1})}{2 \cdot \Delta y}$$

$\mathcal{D}_y =$ 

$$\left(\begin{array}{cccc|cccc|c|cccc}
d_{0,0} & & & & d_{0,1} & & & & & d_{0,N} & & & \\
& d_{0,0} & & & & d_{0,1} & & & & & d_{0,N} & & \\
& & \ddots & & & & \ddots & & & & & \ddots & \\
& & & d_{0,0} & & & & d_{0,1} & & & & & d_{0,N} \\
\hline
d_{1,0} & & & & d_{1,1} & & & & & d_{1,N} & & & \\
& d_{1,0} & & & & d_{1,1} & & & & & d_{1,N} & & \\
& & \ddots & & & & \ddots & & & & & \ddots & \\
& & & d_{1,0} & & & & d_{1,1} & & & & & d_{1,N} \\
\hline
& & & & & & & & \ddots & & & & \\
\hline
d_{N,0} & & & & d_{N,1} & & & & & d_{N,N} & & & \\
& d_{N,0} & & & & d_{N,1} & & & & & d_{N,N} & & \\
& & \ddots & & & & \ddots & & & & & \ddots & \\
& & & d_{N,0} & & & & d_{N,1} & & & & & d_{N,N}
\end{array}\right)$$

PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

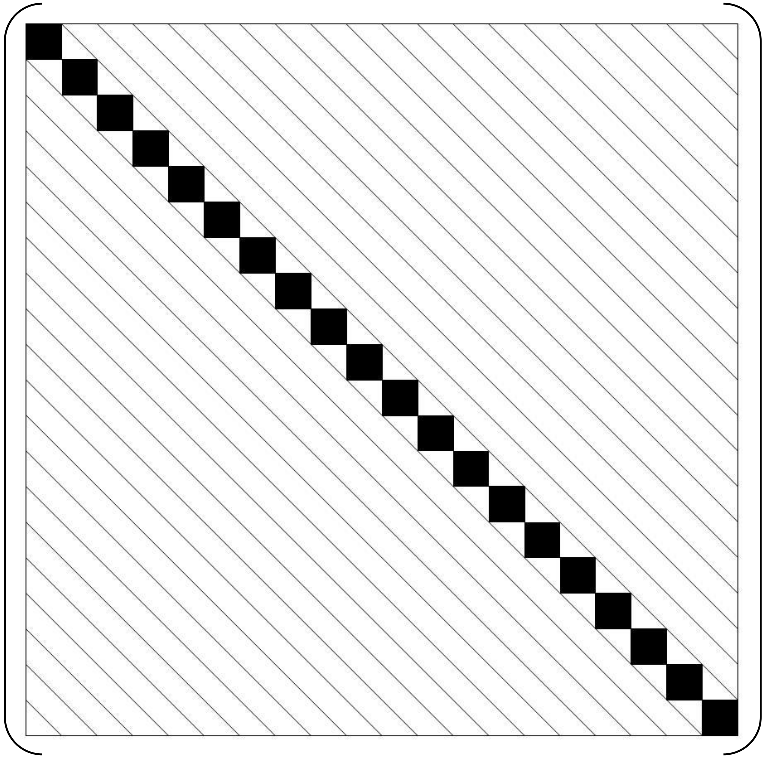
$$
\begin{pmatrix} & & \\ & Dx & \\ & & \end{pmatrix}
\begin{pmatrix} U(0,0) \\ U(1,0) \\ . \\ U(Nx,0) \\ ---- \\ U(0,1) \\ U(1,1) \\ . \\ U(Nx,1) \\ ---- \\ . \\ . \\ ---- \\ U(0,Ny) \\ U(1,Ny) \\ . \\ U(Nx,Ny) \end{pmatrix}
=
\begin{pmatrix} U_x(0,0) \\ U_x(1,0) \\ . \\ U_x(Nx,0) \\ ---- \\ U_x(0,1) \\ U_x(1,1) \\ . \\ U_x(Nx,1) \\ ---- \\ . \\ . \\ ---- \\ U_x(0,Ny) \\ U_x(1,Ny) \\ . \\ U_x(Nx,Ny) \end{pmatrix}
\qquad
\begin{pmatrix} & & \\ & Dy & \\ & & \end{pmatrix}
\begin{pmatrix} U(0,0) \\ U(1,0) \\ . \\ U(Nx,0) \\ ---- \\ U(0,1) \\ U(1,1) \\ . \\ U(Nx,1) \\ ---- \\ . \\ . \\ ---- \\ U(0,Ny) \\ U(1,Ny) \\ . \\ U(Nx,Ny) \end{pmatrix}
=
\begin{pmatrix} U_y(0,0) \\ U_y(1,0) \\ . \\ U_y(Nx,0) \\ ---- \\ U_y(0,1) \\ U_y(1,1) \\ . \\ U_y(Nx,1) \\ ---- \\ . \\ . \\ ---- \\ U_y(0,Ny) \\ U_y(1,Ny) \\ . \\ U_y(Nx,Ny) \end{pmatrix}
$$

PROBLEMA DE CONTORNO: Extensión a 2D

Las matrices de derivación se distribuyen de manera diferente en cada dirección:

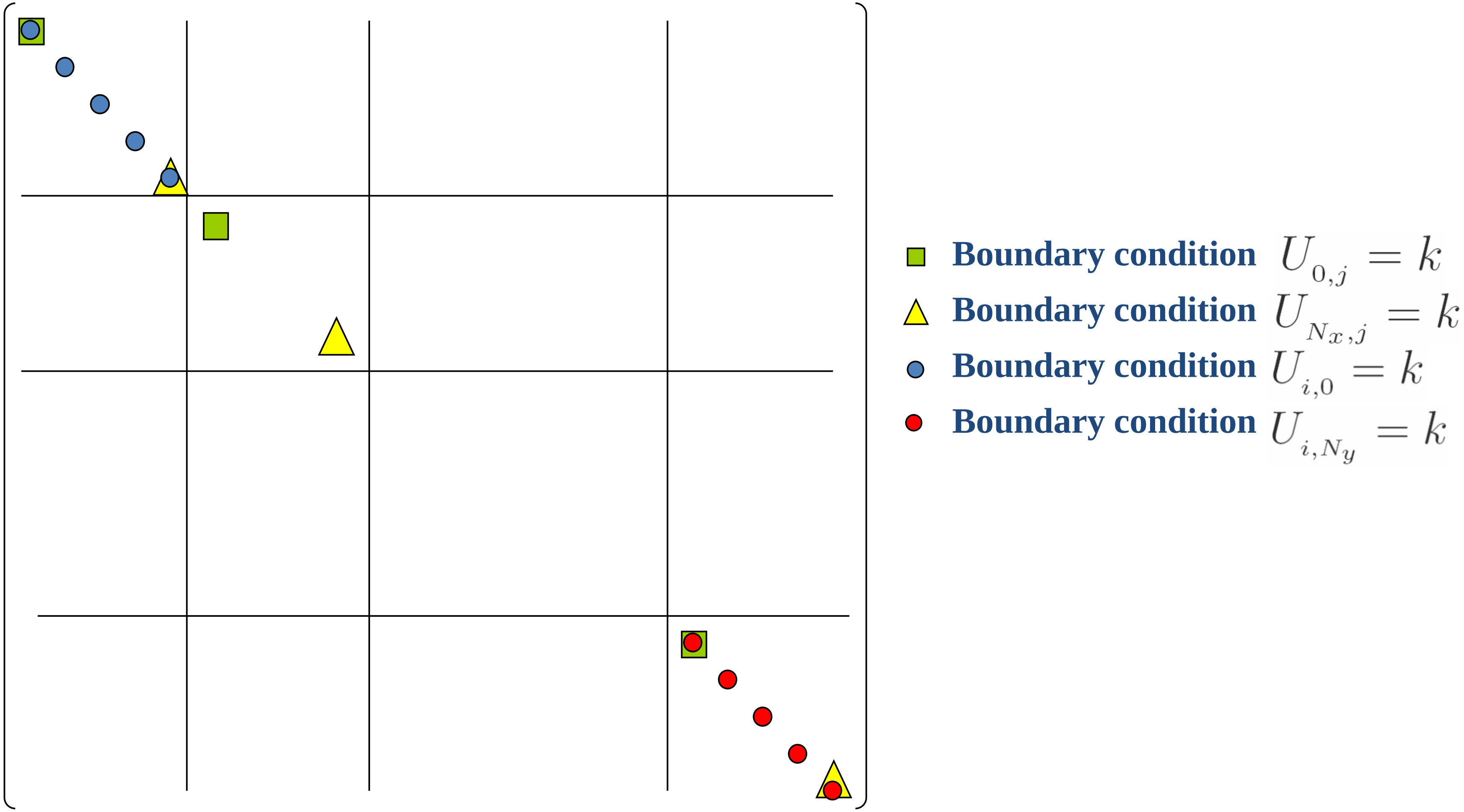
$$\nabla^2 = U_{xx} + U_{yy} \Longrightarrow A = \mathcal{D}_x^2 + \mathcal{D}_y^2$$

$A =$



PROBLEMA DE CONTORNO: Extensión a 2D

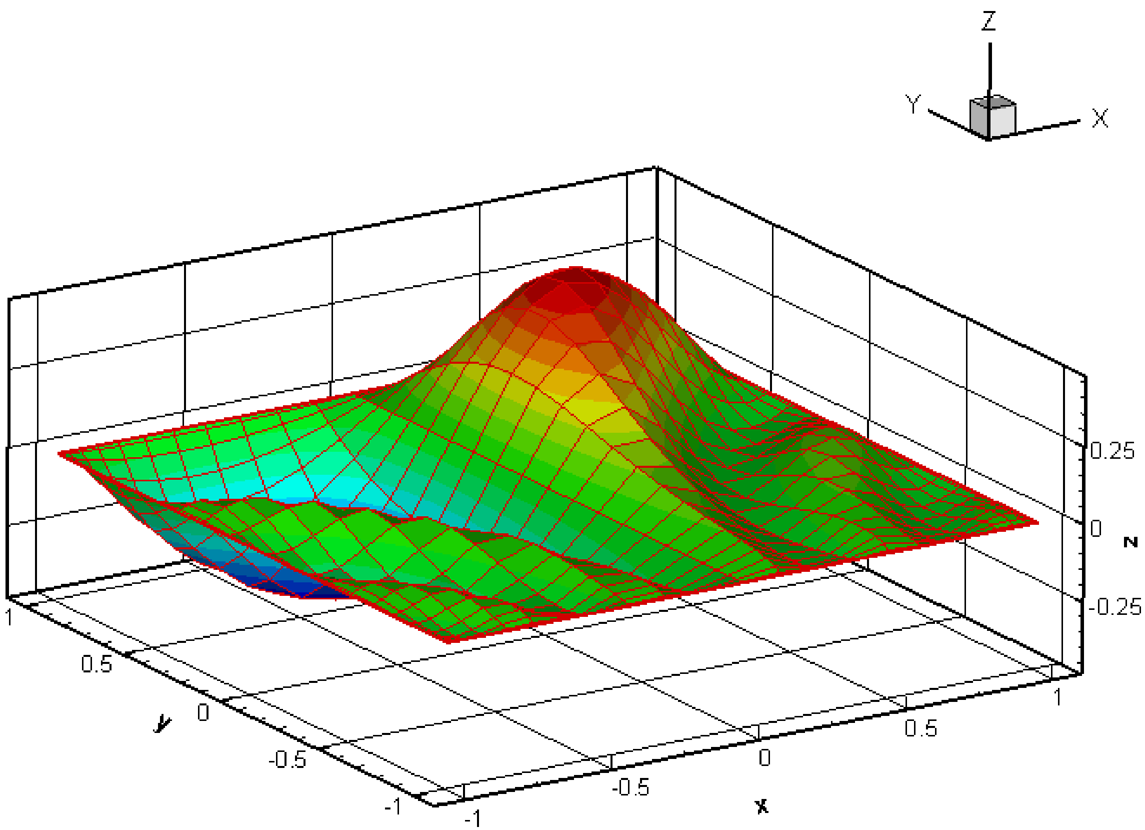
Condiciones de contorno:



PROBLEMA DE CONTORNO: Extensión a 2D

Ejercicio I:

$$\Delta u = 10 \cdot \sin(8 \cdot x \cdot (y-1)) \qquad en\ \Omega = [-1, 1]x[-1, 1]$$
$$u = 0. \qquad en\ \partial\Omega$$



Problema de Valor inicial

Método de Euler

- ✓ Se basa en reescribir el problema de Cauchy usando como aproximación a la derivada una de las fórmulas de derivación numérica.

Problema de Valor inicial

Método de Euler

- ✓ Se basa en reescribir el problema de Cauchy usando como aproximación a la derivada una de las fórmulas de derivación numérica.

Problema de Cauchy

$$\frac{dy}{dt} = f(y); \;\; y(0) = y_0$$

Problema de Valor inicial

Método de Euler

✓ Se basa en reescribir el problema de Cauchy usando como aproximación a la derivada una de las fórmulas de derivación numérica.

Problema de Cauchy

$$\frac{dy}{dt} = f(y); \;\; y(0) = y_0$$

Diferencias atrasadas

$$\frac{dy}{dt} = \frac{y(t)-y(t-h)}{h}$$

METODO DE EULER

- ✓ Se basa en reescribir el problema de Cauchy usando como aproximación a la derivada una de las fórmulas de derivación numérica.

Problema de Cauchy

$$\frac{dy}{dt} = f(y); \quad y(0) = y_0$$

Derivada numérica atrasada

$$\frac{dy}{dt} = \frac{y(t) - y(t-h)}{h}$$

Esquema numérico (Euler)

$$\frac{y(t) - y(t-h)}{h} = f(y(t-h))$$

$$y(t) = y(t-h) + hf(y(t-h))$$

METODO DE EULER

✓ Se basa en reescribir el problema de Cauchy usando como aproximación a la derivada una de las fórmulas de derivación numérica.

Problema de Cauchy

$$\frac{dy}{dt} = f(y); \quad y(0) = y_0$$

Derivada numérica 'retrasada'

$$\frac{dy}{dt} = \frac{y(t) - y(t-h)}{h}$$

Esquema numérico (Euler)

$$\frac{y(t) - y(t-h)}{h} = f(y(t-h))$$

$$y(t) = y(t-h) + hf(y(t-h))$$

✓ Es necesario discretizar la variable independiente *t* en pasos de ancho *h:*

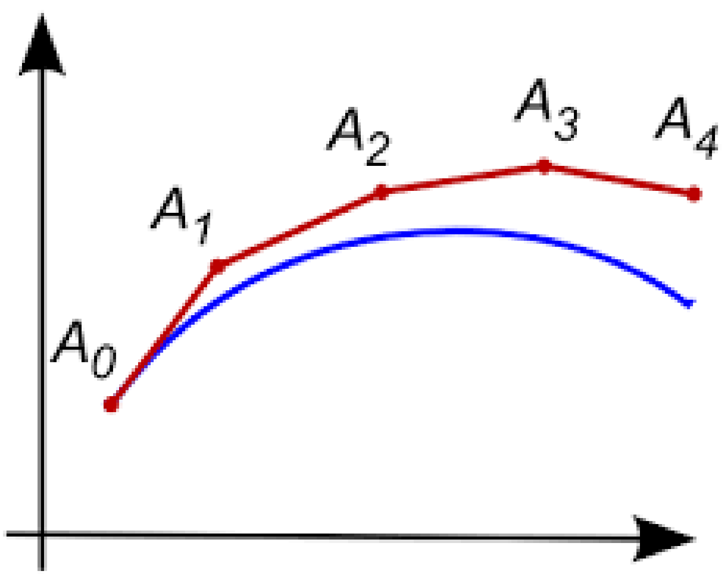
(i) Si t no está acotado, simplemente definimos un valor de h.

(ii) Si t ϵ [a,b], entonces $y(a) = y_0$, y h se define a partir del número de subintervalos *n* en el que discretizamos [a,b] tal que $\mathrm{h} = \frac{b-a}{n}$

METODO DE EULER

Ejemplo:

$$\begin{cases} \dfrac{dU(t)}{dt} = -U(t) \\ U(0) = 1 \end{cases}$$



Llamando:

$$t_n = t_0 + n \cdot dt; \quad U^n = U(t_n); \quad F(U^n) = -U^n;$$

y tomando:

$$\frac{dU}{dt} = \frac{U^{n+1} - U^n}{dt}$$

Resulta la expresión:

$$\frac{U^{n+1} - U^n}{dt} = -U^n \rightarrow U^{n+1} = U^n - dt \cdot U^n$$

De modo iterativo desde $U(t_0) = U(0) = U^0 = 1$

$$U^1 = U^0 - dt \cdot U^0 = U^0(1 - dt)$$

$$U^2 = U^1 - dt \cdot U^1 = U^1(1 - dt) = U^0(1 - dt)^2$$

$$U^3 = U^2 - dt \cdot U^2 = U^2(1 - dt) = U^0(1 - dt)^3$$

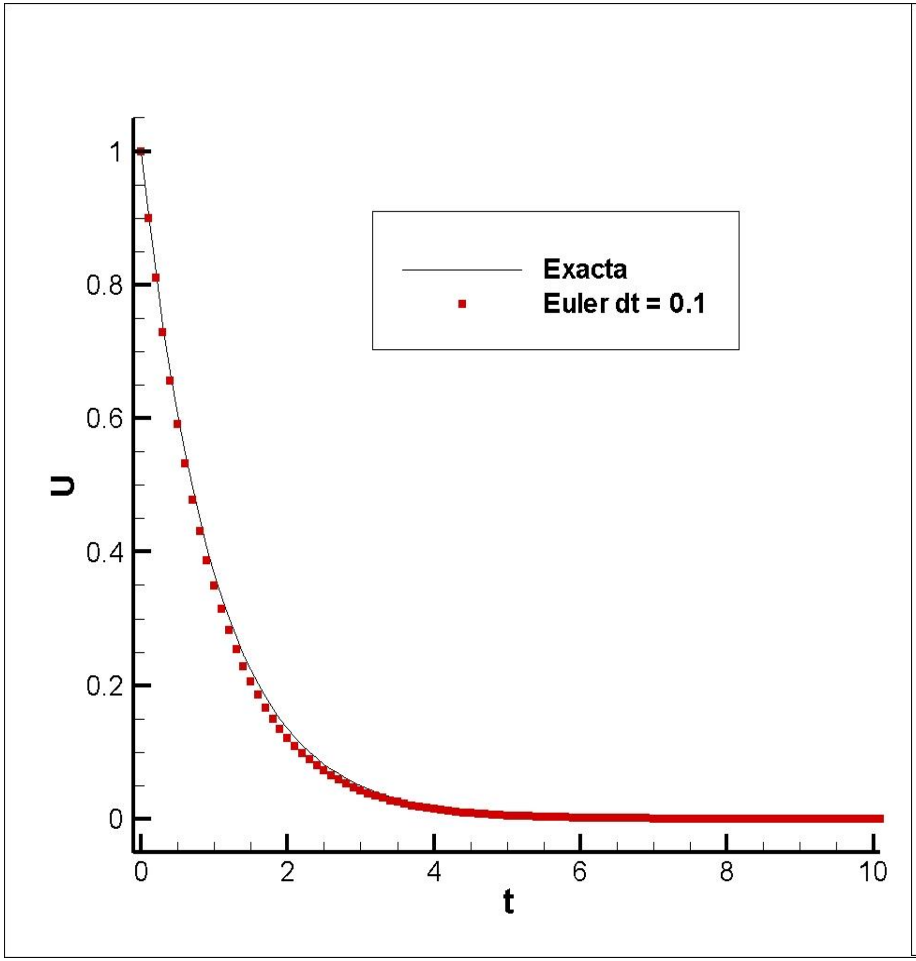
...

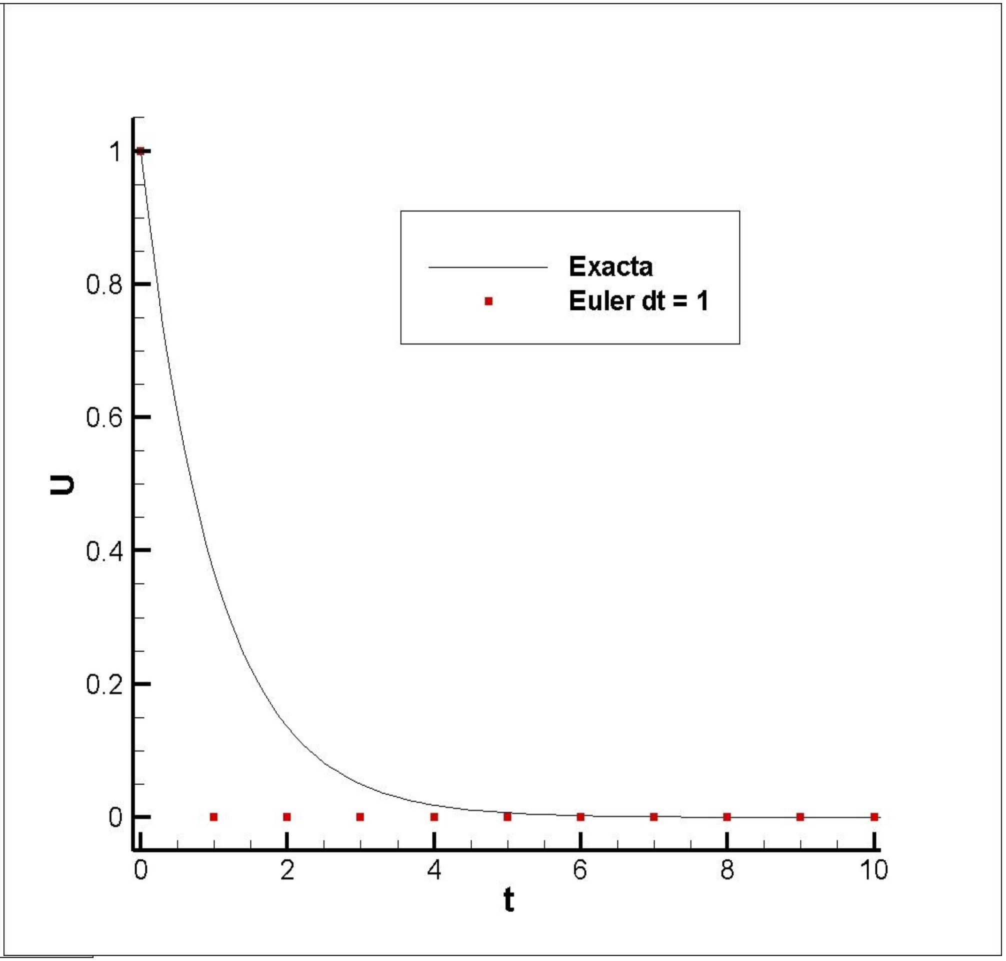
$$U^{n+1} = U^n - dt \cdot U^n = U^n(1 - dt) = U^0(1 - dt)^{n+1}$$

METODO DE EULER

Ejemplo:

$$\begin{cases} \dfrac{dU(t)}{dt} = -U(t) \\ U(0) = 1 \end{cases}$$





CLASIFICACION DE LOS METODOS NUMERICOS

❖ Esquemas explícitos vs esquemas implícitos:

Si en el algoritmo que discretiza la EDO, el valor de la variable en el paso (k+1)-ésimo, $U^{(k+1)}$ puede expresarse únicamente en función de la información en pasos anteriores, se dice que el esquema es ***explícito:***

$$U^{(k+1)} = \boldsymbol{f}\left(U^{(k)}, U^{(k-1)}, \dots\right)$$

Si $U^{(k+1)}$ no se puede despejar explícitamente en función de los pasos anteriores, diremos que el esquema es ***implícito.***

Ejemplos: en derivación numérica, el esquema de diferencias **retrasadas** siempre es explícito, mientras que el de diferencias **centradas** es implícito.

La implementación de un esquema explícito es, por tanto, mucho más sencilla que la de un esquema implícito en general.

CLASIFICACION DE LOS METODOS NUMERICOS

❖ <u>Ejemplo de esquema implicito. Backward Euler.</u>

Dado un problema de valor inicial,

$$\frac{dy}{dt} = f(t, y)$$

evaluamos la derivada usando diferencias finitas, como en un esquema explícito. Sin embargo, evaluamos la función en el instante siguiente:

$$y_{k+1} = y_k + hf(t_{k+1}, y_{k+1}).$$

El valor de la función en el instante siguiente, $f(t_{k+1},y_{k+1})$ no es conocido. Es necesario encontrar el valor y_{k+1} que anula la función:

$$G = y_{k+1} - y_k - hf(t_{k+1}, y_{k+1})$$

Este valor lo podemos encontrar utilizando el método de Newton en cada iteración sobre la función G:

$$y_{k+1}^{i+1} = y_{k+1}^{i} - \frac{G(y_{k+1}^{i})}{G'(y_{k+1}^{i})} \qquad G'\left(y_{k+1}^{i}\right) = 1 - h\left.\frac{df}{dy}\right]_{(t_{k+1}, y_{k+1}^{i})}$$

CLASIFICACION DE LOS METODOS NUMERICOS

❖ Esquemas unipaso vs esquemas multipaso:

Los esquemas **unipaso** son aquellos en donde para la determinación del valor de $U^{(k+1)}$ necesitamos conocer únicamente el valor de $U^{(k)}$ y de $F(U^{(k)})$. Por tanto el esquema numérico que describe la EDO consta de una ecuación y una única condición inicial.

Los esquemas **multipaso** son aquellos en donde para la determinación del valor de $U^{(k+1)}$ necesitamos conocer el valor de U ,y de $F(U)$ en p(p>1) pasos anteriores, y necesitamos ir almacenando esa información en diferentes pasos. Por tanto estos métodos requieren p condiciones iniciales.

Por sencillez, en este curso nos centraremos en esquemas unipaso de tipo explícito.

METODO DE RUNGE-KUTTA

- ✓ La idea de este método es ‘mejorar’ el Euler, no tanto usando aproximaciones a la discretización de la derivada de orden superior (término de la izquierda en el problema original), sino afinando en la particularización de la función f (término de la derecha en el problema original).

$$y_{i+1} = y_i + \underbrace{h\phi_n(t_i, y_i, h)}_{\text{En lugar de poner } f(y_i)}$$

$$\boxed{y_{i+1} = y_i + h\phi_n(t_i, y_i, h)}$$

$$\phi_n = a_1 k_1 + a_2 k_2 + \cdots + a_n k_n$$

$$\begin{aligned}
k_1 &= f(t_i, y_i) \\
k_2 &= f(t_i + p_1 h, y_i + q_{11} k_1 h) \\
k_3 &= f(t_i + p_2 h, y_i + q_{21} k_1 h + q_{22} k_2 h) \\
&\vdots \\
k_n &= f(t_i + p_{n-1} h, y_i + q_{n-1,1} k_1 h + q_{n-1,2} k_2 h + \cdots + q_{n-1,n-1} k_{n-1} h)
\end{aligned}$$

Las constantes {p’s, q’s, a’s} se calculan de tal forma que el método Runge-Kutta de orden *n* coincida, hasta orden h^n, con el método de Taylor de orden *n.*

De esta forma, los métodos Runge-Kutta alcanzan la misma precisión que los Taylor, pero son mucho más fáciles de implementar (sustituyen el cálculo explícito de las derivadas de f por las evaluaciones funcionales de f.

- ✓ El RK de orden 1 coincide con el método de Euler.

- ✓ Existen diferentes tipos de RK de orden 2, donde la siguiente relación entre coeficientes ha de cumplirse

$$\begin{aligned} a_1 + a_2 &= 1 \\ a_2 p_1 &= \frac{1}{2} \\ a_2 q_{11} &= \frac{1}{2} \end{aligned} \tag{6.27}$$

Destacan tres tipos concretos de RK2, que son:

Método de Heun. Si en la primera de las ecuaciones (6.27) elegimos $a_1 = a_2 = 1/2$, entonces $p_1 = q_{11} = 1$. Llevados estos resultados a (6.24) y posteriormente a (6.23) y (6.22), esta última resulta

$$
\begin{aligned}
y_{i+1} &= y_i + \frac{h}{2}(k_1 + k_2) \\
k_1 &= f(t_i, y_i) \\
k_2 &= f(t_i + h, y_i + hk_1)
\end{aligned}
$$

que es el llamado *método de Heun* o *método modificado de Euler*. La pendiente re-

Método del punto medio. Otra posible elección para la solución de (6.27) sería $a_1 = 0$, $a_2 = 1$, $p_1 = q_{11} = 1/2$. De esta forma, resulta

$$
\begin{aligned}
y_{i+1} &= y_i + hk_2 \\
k_1 &= f(t_i, y_i) \\
k_2 &= f(t_i + \frac{h}{2}, y_i + \frac{h}{2}k_1)
\end{aligned}
$$

Método de Ralston. Veamos por último, otra posible solución de (6.27) como $a_1 = 1/3$, $a_2 = 2/3$, $p_1 = q_{11} = 3/4$. De esta forma, se obtiene

$$
\begin{aligned}
y_{i+1} &= y_i + \frac{h}{3}(k_1 + 2k_2) \\
k_1 &= f(t_i, y_i) \\
k_2 &= f(t_i + \frac{3}{4}h, y_i + \frac{3}{4}hk_1)
\end{aligned}
$$

Runge-Kutta de orden 4. Los métodos Runge-Kutta más utilizados son los de orden 4. Los cálculos son muy complicados y sólamente expondremos una posible elección

$$
\begin{aligned}
y_{i+1} &= y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\
k_1 &= f(t_i, y_i) \\
k_1 &= f(t_i + \frac{h}{2}, y_i + \frac{h}{2}k_1) \\
k_3 &= f(t_i + \frac{h}{2}, y_i + \frac{h}{2}k_2) \\
k_4 &= f(t_i + h, y_i + hk_3)
\end{aligned}
\tag{6.32}
$$

que es el *método clásico de Runge-Kutta de orden* 4, con error local $\tau_{i+1} = O(h^5)$ y global $\varepsilon_{i+1} = O(h^4)$.

ECUACIONES DIFERENCIALES ORDINARIAS DE ORDEN SUPERIOR

$$\begin{cases} U''(t) + U(t) = 0 \\ U(0) = 1 \\ U'(0) = 0 \end{cases}$$

Orden 2 (derivada segunda)
Necesita 2 condiciones iniciales: función y derivada

Convertimos el problema de orden 2 en un sistema de orden 1.

$$\vec{V}(t) = \begin{pmatrix} V_1(t) \\ V_2(t) \end{pmatrix} = \begin{pmatrix} U(t) \\ U'(t) \end{pmatrix}$$

Resultando:

$$\frac{d\vec{V}(t)}{dt} = \begin{pmatrix} V_1'(t) \\ V_2'(t) \end{pmatrix} = \begin{pmatrix} U'(t) \\ U''(t) \end{pmatrix}$$

Despejando de la ecuación original

$$\frac{d\vec{V}(t)}{dt} = \begin{pmatrix} V_1'(t) \\ V_2'(t) \end{pmatrix} = \begin{pmatrix} U'(t) \\ U''(t) \end{pmatrix} = \begin{pmatrix} U'(t) \\ -U(t) \end{pmatrix} = \begin{pmatrix} V_2(t) \\ -V_1(t) \end{pmatrix} = \vec{F}(\vec{V})$$

ECUACIONES DIFERENCIALES ORDINARIAS DE ORDEN SUPERIOR

$$\begin{cases} U''(t) + U(t) = 0 \\ U(0) = 1 \\ U'(0) = 0 \end{cases}$$

Orden 2 (derivada segunda)
Necesita 2 condiciones iniciales: función y derivada

El problema se completa con las condiciones de contorno de las nuevas variables.

$$\begin{cases} \dfrac{d\vec{V}(t)}{dt} = \begin{pmatrix} V_1'(t) \\ V_2'(t) \end{pmatrix} = \begin{pmatrix} V_2(t) \\ -V_1(t) \end{pmatrix} = \vec{F}(\vec{V}) \\ \vec{V}(0) = \begin{pmatrix} V_1(0) \\ V_2(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{cases}$$

METODO DE EULER

Implementación de la subrutina del método

```
!****************************************************************
!*                                                              *
!*  Euler method for the Initial Value Problem :                *
!*                   dU/dt = F(U,t),     U(t_0) = U^0           *
!*                                                              *
!*  Numerical scheme :                                          *
!*                   U^{n+1} = U^n  +  dt * F(U^n,t_n)          *
!*                                                              *
!*  (explicit and first order accurate)                         *
!*                                                              *
!*                                                              *
!*                                                              *
!*   Inputs :                                                   *
!*                                                              *
!*           m  : system of m equations                         *
!*           U  : m--dimensional vector U(t_n)                  *
!*           t  : time                                          *
!*           F  : m--dimensional vector valued function F(U(t_n), t_n)  *
!*          dt  : time step                                     *
!*                                                              *
!*                                                              *
!*                                                              *
!*  Outputs :                                                   *
!*                                                              *
!*           U  : m--dimensional vector U(t_{n+1})              *
!*                                                              *
!*                                                              *
!*                                                              *
!*                                                              *
!****************************************************************
```

```
subroutine Euler  ( m, U, t, dt, F )


    !*** Interface for the m--dimensional vector valued function F
    interface

          function F ( m, U, t )
                integer, intent(in)   :: m
                real(8), intent(in)   :: t, U(1:m)

                real(8)   :: F (1:m)
          end function F

    end interface


    !*** Dummy arguments specification
    integer, intent(in)    :: m
    real(8), intent(in)    :: t, dt
    real(8), intent(inout) :: U(1:m)


    !***    U^{n+1} = U^n  +  dt * F ( U^n, t_n )
             U      = U    +  dt * F ( m, U, t )


end subroutine Euler
```

PROBLEMA DE VALOR INICIAL

Ejercicio II: Tiro parabolico

2º Ley de Newton: $\vec{F} = m\vec{a} \rightarrow \begin{cases} F_x = m\vec{a}_x \\ F_y = m\vec{a}_y \end{cases}$

Posición del objeto: $\vec{r}(t) = (x(t), y(t)) \equiv$ Trayectoria en el tiempo.

Vector velocidad: $\vec{r}\,'(t) = (x'(t), y'(t)) \equiv$ Tangente a la trayectoria.

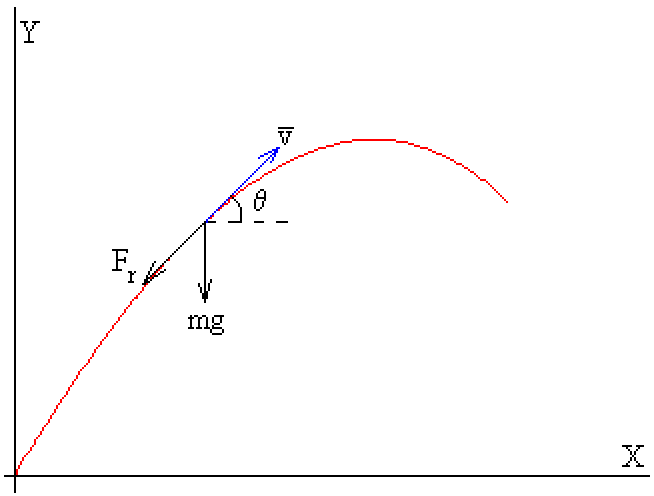
$$\begin{cases} F_x = ma_x \\ F_y = ma_y \end{cases} \rightarrow \begin{cases} m \cdot x''(t) = -F_x roz \\ m \cdot y''(t) = -F_y roz - m \cdot g \end{cases}$$

PROBLEMA DE VALOR INICIAL

Ejercicio II: Tiro parabolico

$$\begin{cases} F_x = ma_x \\ F_y = ma_y \end{cases} \rightarrow \begin{cases} m \cdot x''(t) = -F_x roz \\ m \cdot y''(t) = -F_y roz - m \cdot g \end{cases} \Rightarrow$$

$$\Rightarrow \begin{cases} m \cdot x''(t) = -\dfrac{1}{2} \rho \sqrt{\left(x'(t)\right)^2 + \left(y'(t)\right)^2}\, x'(t) \cdot C_D \cdot S \\ m \cdot y''(t) = -\dfrac{1}{2} \rho \sqrt{\left(x'(t)\right)^2 + \left(y'(t)\right)^2}\, y'(t) \cdot C_D \cdot S - m \cdot g \end{cases}$$



Resistencia total [editar]

La fórmula de la resistencia aerodinámica total creada por un avión en vuelo es:

$$D = qSC_D = \frac{1}{2}\rho V^2 SC_D$$

Donde:

- D - Resistencia. Se utiliza la "D" por el término inglés *drag* (arrastre).
- ρ - Densidad del fluido.
- V - Velocidad.
- S - Superficie alar en planta.
- C_D - Coeficiente aerodinámico de resistencia.
- $q = \frac{1}{2}\rho V^2$ - Este término se denomina presión dinámica.

Por lo tanto, la fórmula del coeficiente aerodinámico de resistencia es:

$$C_D = \frac{D}{\frac{1}{2}\rho V^2 S}$$

PROBLEMA DE VALOR INICIAL

Ejercicio II: Tiro parabólico

$$\begin{cases} m\cdot x''(t) = -\dfrac{1}{2}\rho\sqrt{\left(x'(t)\right)^2+\left(y'(t)\right)^2}\,x'(t)\cdot C_D\cdot S \\ m\cdot y''(t) = -\dfrac{1}{2}\rho\sqrt{\left(x'(t)\right)^2+\left(y'(t)\right)^2}\,y'(t)\cdot C_D\cdot S - m\cdot g \end{cases}$$

Resolución numérica:

$$\vec{U}(t) = \begin{pmatrix} U_1(t) \\ U_2(t) \\ U_3(t) \\ U_4(t) \end{pmatrix} = \begin{pmatrix} x(t) \\ x'(t) \\ y(t) \\ y'(t) \end{pmatrix} \qquad \frac{d\vec{U}(t)}{dt} = F(\vec{U}(t)) = \begin{pmatrix} x'(t) \\ \ldots despejando\ldots \\ y'(t) \\ \ldots despejando\ldots \end{pmatrix}$$

Condiciones iniciales: Posición inicial: (x(0), y(0))

Velocidad inicial: (x’(0),y’(0))

Resolución: Euler – RK4

Problema (para casa)

Implementa el RK4 y empléalo para resolver el problema de Cauchy anteriormente propuesto. Compara los resultados con los arrojados por el método de Euler y con la solución analítica.

TEMAS FUERA DEL TEMARIO

- ✓ **Convergencia: consistencia y estabilidad**