Preliminares matemáticos

Facultad de Ciencias de la Electrónica

Benemérita Universidad Autónoma de Puebla

Licenciatura en Ingeniería Mecatrónica

Dr. Fernando Reyes Cortés

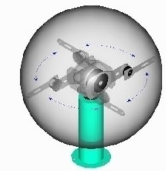
Robótica

ftp://ece.buap.mx/pub/profesor/FernandoReyes/Robotica/

Primavera 2020

Parte VII

Diferenciación numérica



Robótica FCE

BUAP

Contenido

Método de Euler

Un método ampliamente utilizado en robótica y sistemas mecatrónicos es la integración numérica discreta, la cual se puede establecer directamente del método de Euler.

Por ejemplo, al derivar con respecto al tiempo la integral $I(t)$ de la función $f(t)$ se obtiene lo siguiente:

$$I(t) = \int_{t_1}^{t_2} f(\sigma) d\sigma \ \Rightarrow \ \dot{I}(t) = f(t) \ \ \forall t \in [t_1, t_2]. \tag{1}$$

Partiendo de la definición matemática de la derivada se tiene:

$$\dot{I}(t) \ = \ \lim_{\triangle t \to 0} \frac{I(t + \triangle t) - I(t)}{\triangle t} \tag{2}$$

$$\tag{3}$$

donde $\triangle t = t_2 - t_1$, representa un infinitésimo intervalo de tiempo.

Si aproximamos a la derivada $\dot{I}(t)$ en su forma discreta $\dot{I}(t) \simeq \dot{I}(t_k)$ entonces el tiempo discreto está dado por $t_k = kh$, $k = 1, 2, \cdots, n$ y h es el periodo de muestreo. De esta forma se cumple $t_{k-1} = (k-1)h$ y $\triangle t_k = t_k - t_{k-1} = kh - (k-1)h = h$.

Por lo tanto, la derivada discreta $\dot{I}(t_k)$ se puede realizar por diferenciación numérica de la posición $I(t_k)$ de la siguiente forma:

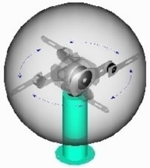
$$\dot{I}(t_k) = \frac{I(t_k) - I(t_{k-1})}{t_k - t_{k-1}} = \frac{I(t_k) - I(t_{k-1})}{h} = f(t_k). \tag{4}$$

La expresión 4 se conoce como método de Euler y sirve para estimar la velocidad por diferenciación numérica de la posición.

Despejando $I(t_k)$ de la expresión (4), la integral discreta adquiere la siguiente forma:

$$I(t_k) = I(t_{k-1}) + hf(t_k) \tag{5}$$

La expresión (5) es muy simple, se convierte en una sumatoria y es adecuada para poderse implementar como algoritmo recursivo.



Robótica FCE

BUAP

Diferenciación numérica

Diferenciación numérica

Es la técnica de aproximar a la derivada de una función por métodos numéricos.

La derivada es una operación matemática sobre funciones o variables de estado, y que tiene una diversidad de aplicaciones en ingeniería.

- La derivada de una función representa la razón de cambio con respecto al tiempo: $\frac{d}{dt}f(t)$.
- Se utilizan en esquemas de control y en procesos de automatización:

$$\boldsymbol{\tau} \;=\; K_p\tilde{\boldsymbol{q}} + K_i\int_0^t \tilde{\boldsymbol{q}}(\sigma)d\sigma - K_v\frac{d}{dt}\boldsymbol{q}(t)$$

- Los algoritmos incluyen un término denominado acción de control derivativo con la finalidad de generar amortiguamiento o freno mecánico y con dicha acción se pretende mejorar la respuesta transitoria del robot. Para lograr esto, se inyecta la velocidad de movimiento, la cual representa la variación temporal de la posición.

Robótica FCE

BUAP

- La fricción viscosa y de Coulomb son fenómenos físicos que se encuentran presentes en los sistemas mecánicos, la derivada de la posición forma parte de este fenómeno: $B\dot{\boldsymbol{q}} + F_c \text{signo}(\dot{\boldsymbol{q}})$.

- El modelado dinámico de sistemas mecatrónicos y robots incluye derivadas de las variables de estado para conformar la estructura matemática adecuada para estudiar y analizar todos los fenómenos físicos del sistema: $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$.

- La teoría de estabilidad de sistemas dinámicos es otro ejemplo donde se utilizan derivadas: $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$; $V(\boldsymbol{x}) > 0 \Longrightarrow \dot{V}(\boldsymbol{x}) < 0)$.

- La derivada de la energía (potencia) se emplea para llevar a cabo el análisis de estabilidad del punto de equilibrio: $\dot{V}(\boldsymbol{x}) < 0)$

- Hoy en día, la derivada se ha convertido en una herramienta imprescindible para realizar automatización de sistemas.

La notación de la derivada de una función $f(t)$ con respecto al tiempo es definida como $\dot{f}(t)$, la cual es igual a la razón de cambio de f con respecto al tiempo t. Matemáticamente la derivada de una función se define como:

$$\dot{f}(t) \;=\; \frac{d}{dt}f(t) = \lim_{\triangle t\to 0}\frac{f(t+\triangle t)-f(t)}{\triangle t} \tag{6}$$

donde $\triangle t$ es un infinitésimo intervalo de tiempo.

En mecatrónica hay varios procesos donde se requiere medir la razón de cambio de las variables del robot. Por ejemplo,

- La razón de cambio de la posición es la velocidad de movimiento.
- El cambio temporal de la velocidad es la aceleración.

Por otro lado, la integral de la aceleración es la velocidad, y la integral de la velocidad es la posición. Es decir, existe una relación muy cercana entre la integral y la derivada; se puede considerar que son operaciones inversas una de la otra. La integral de una derivada:

$$\int_0^t \dot{f}(\sigma)d\sigma \;=\; \int_0^t \frac{df(\sigma)}{d\sigma}d\sigma = \int_0^t df(\sigma) = f(t)\Big|_0^t = f(t)-f(0)$$

retorna la función original $f(t)$ más una constante que depende de las condiciones iniciales.

La derivada de una integral

$$\frac{d}{dt}\int_0^t f(\sigma)d\sigma = f(t)$$

retorna la función original.

Geométricamente, la derivada $\dot{f}$ puede ser descrita como la pendiente de la línea tangente en la función $f(t)$ del punto específico t. La línea tangente está especificada por la pendiente $\frac{f(t+\triangle t)-f(t)}{\triangle t}$ (ver figura 2.)

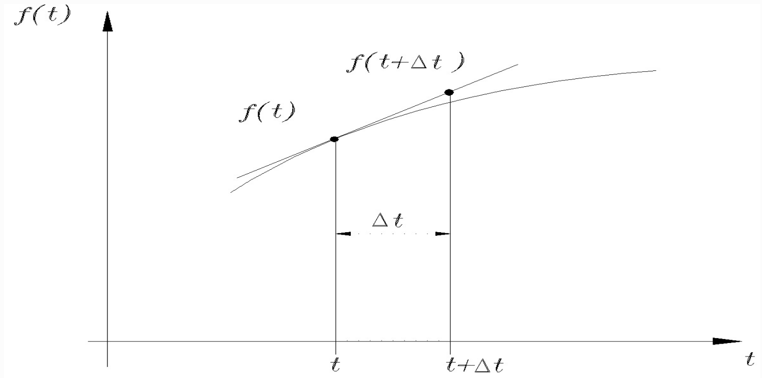


Figura 1: Línea tangente de la función f en el punto t_k.

Debido a esto, la derivada de una constante es cero, puesto que la línea tangente es horizontal (no hay variación temporal). Los puntos donde la derivada de la función f es cero se llaman ***puntos críticos*** y pueden representar regiones horizontales de la función f o ***puntos extremos*** (puntos que son máximo o mínimo local o también globales).

Las técnicas de diferenciación numérica estiman la derivada de una función f en un punto t_k aproxima la pendiente de la línea tangente en t_k usando valores de la función en puntos cercanos a t_k. Si denotamos al intervalo de tiempo $\triangle t$ como la diferencia entre dos puntos consecutivos, $\triangle t = t_k - t_{k-1} = h$, donde h es la longitud de $\triangle t$, (ver figura 2)

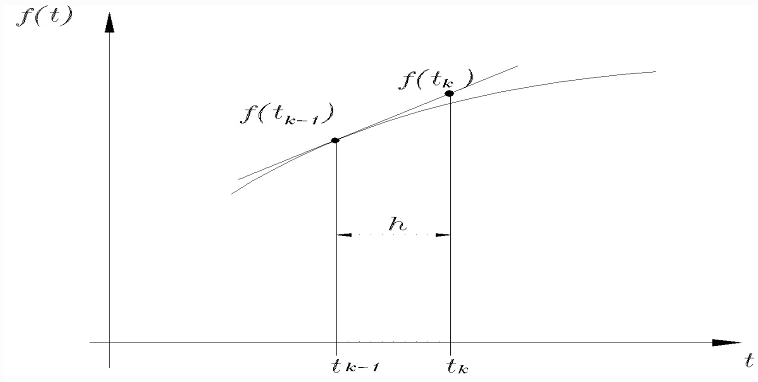


Figura 2: Línea tangente de la función f en el punto t_k.

En control digital y sistemas discretos la longitud del intervalo h se mantiene constante (periodo de muestreo) y se toman datos cada determinado tiempo h, igualmente espaciadas, esto significa que el tiempo discreto transcurre como múltiplos del periodo de muestreo $t_k = t + h$; para n muestras se tiene que el tiempo discreto se puede expresar como $t_k = kh$, donde $k = 1, 2, 3, \cdots, n$.

Una forma ampliamente usada es el método de Euler (diferenciación numérica)

$$\dot{f}(t_k) \simeq \frac{f(t_k) - f(t_{k-1})}{t_k - t_{k-1}} = \frac{f(t_k) - f(t_{k-1})}{h}$$

particularmente esta aproximación es conocida como diferenciación con un paso atrás (*backward difference*).

También se puede obtener la aproximación de la derivada por computar la pendiente entre $f(t_k)$ y $f(t_{k+1})$ conocida como diferenciación hacia adelante (*forward difference*):

$$\dot{f}(t_k) \simeq \frac{f(t_{k+1}) - f(t_k)}{t_{k+1} - t_k} = \frac{f(t_{k+1}) - f(t_k)}{h}$$

la figura 3 ilustra ambos métodos.

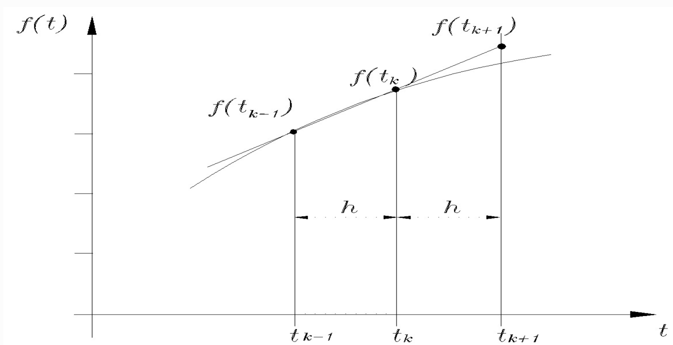


Figura 3: Diferenciación numérica: *backward* y *forward*.

Robótica FCE

BUAP

MATLAB: diferenciación numérica

Función diff

MATLAB tiene la función **diff** la cual realiza la diferenciación numérica entre valores adyacentes de un vector x. La sintaxis de la función **diff** es la siguiente:

y= `diff(`x`)`

- **diff** — Retorna un nuevo vector y conteniendo la diferenciación entre valores adyacentes del vector de entrada x

La función **diff** también se aplica a matrices, entonces opera en cada columna de la matriz, y retorna una matriz con el mismo número de columnas, en este caso la sintaxis es:

B= `diff(`A`)`

- **diff** — Retorna una nueva matriz B conteniendo la diferenciación entre valores adyacentes de cada columna de la matriz de entrada A.

Ejemplo 4.1:

Aproximar la derivada con respecto al tiempo de la función $\operatorname{sen}(t)$ mediante diferenciación.

Para obtener la derivada temporal por diferenciación numérica de la función $\operatorname{sen}(t)$ se hace uso de **diff**. El intervalo de diferenciación numérica se selecciona de 0 a 10 segundos, con incrementos de un milisegundo. El método analítico conduce $\frac{d}{dt}\operatorname{sen}(t) = \cos(t)$. Por lo tanto, se comparará la solución aproximada con la solución analítica.

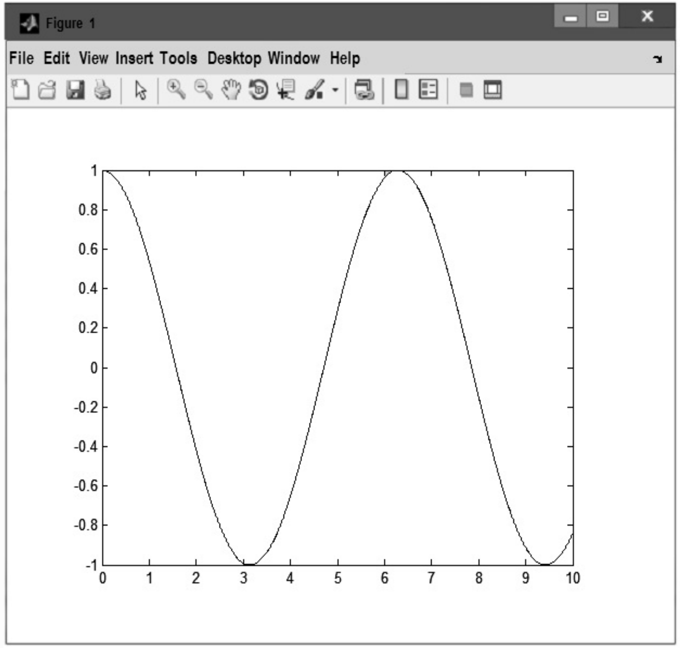


Figura 4: Comparación del método analítico y por diferenciación numérica.

Código MATLAB 2: derivada_seno

Robótica, Período de primavera (24 de enero de 2020). Licenciatura en Ingeniería Mecatrónica.
Dr. Fernando Reyes Cortés. Facultad de Ciencias de la Electrónica, BUAP.

derivada_seno.m — MATLAB 2019a

```
clc;
clear all;
close all;
 % intervalo de tiempo
t=0:0.001:10;
 % función a derivar
f=sin(t);
 % derivada de la función sen(t) con respecto al tiempo.
 % es decir se obtiene $\dot{f}(t_k) = \frac{df(t_k)}{dt_k}$
dfdt = diff(f)./diff(t);
 % el vector df tiene dimensión n-1
t1=0:0.001:9.999; % nueva base de tiempo con dimensión n-1
 %compara la derivada aproximada con el método analítico
plot(t1,dfdt,t1,cos(t1))
```

Robótica FCE
BUAP

Ecuaciones diferenciales lineales: solución numérica

La calidad de la derivada depende de la distancia entre esos puntos o del periodo de muestreo $h = t_k - t_{k-1} = t_{k+1} - t_k$. De esta forma se puede obtener la aceleración por diferenciación numérica de la velocidad:

$$\ddot{f}(t_k) = \frac{\dot{f}(t_k) - \dot{f}(t_{k-1})}{h} = \frac{f(t_k) - 2f(t_{k-1}) + f(t_{k-2})}{h^2}$$

Ejemplo 5.1:

Emplear el método de Euler para convertir la siguiente ecuación diferencial en su equivalente expresión por diferenciación numérica.

$$\alpha u(t) = a_2\ddot{y}(t) + a_1\dot{y}(t) + a_0 y(t) \tag{7}$$

Empleando el método de Euler, la velocidad y aceleración toman la siguiente forma:

$$\begin{aligned}
\dot{y}(t) &\simeq \dot{y}(t_k) = \frac{y(t_k) - y(t_{k-1})}{h} \\
\ddot{y}(t) &\simeq \ddot{y}(t_k) = \frac{\dot{y}(t_k) - \dot{y}(t_{k-1})}{h} = \frac{\frac{y(t_k)-y(t_{k-1})}{h} - \frac{y(t_{k-1})-y(t_{k-2})}{h}}{h} \\
\ddot{y}(t_k) &= \frac{y(t_k) - 2y(t_{k-1}) + y(t_{k-2})}{h^2}
\end{aligned}$$

Robótica FCE

BUAP

Por lo tanto, la ecuación diferencial 7 queda como:

$$\begin{aligned}\alpha u(t_k) &= a_2\ddot{y}(t_k) + a_1\dot{y}(t_k) + a_0 y(t_k)\\ \alpha u(t_k) &= a_2\left[\frac{y(t_k)-2y(t_{k-1})+y(t_{k-2})}{h^2}\right] + a_1\frac{y(t_k)-y(t_{k-1})}{h} + a_0 y(t_k).\end{aligned}$$

La solución para $y(t_k)$ está dada por:

$$y(t_k) = \left[a_0 + \frac{a_1}{h} + \frac{a_2}{h^2}\right]^{-1}\left[\alpha u(t_k) + \left[\frac{2a_2}{h^2} + \frac{a_1}{h}\right] y(t_{k-1}) - \frac{a_2}{h^2} y(t_{k-2})\right].$$

Para implementar la solución discreta $y(t_k)$ de la ecuación (7) se requieren los estados $y(t_{k-1})$ y $y(t_{k-2})$ y el periodo de muestre h.

El método de Euler es ampliamente utilizado para resolver ecuaciones diferenciales lineales, usando el procedimiento de este ejemplo se obtiene la solución en forma discreta de sistemas dinámicos lineales.

Robótica FCE
BUAP

Código MATLAB 2: simu_sistema_segundo_orden_AB

Robótica, Período de primavera (24 de enero de 2020). Licenciatura en Ingeniería Mecatrónica.
Dr. Fernando Reyes Cortés. Facultad de Ciencias de la Electrónica, BUAP.

simu_sistema_segundo_orden_AB.m MATLAB 2019a

```
clc; clear all; close all; format short
ti=0; tf = 10; h=0.001; % parámetros de simulación
ts=ti:h:tf; % intervalo de simulación.
cond_iniciales=[0;0];
opciones=odeset('RelTol' ,1e-06, 'AbsTol' ,1e-06,'InitialStep' ,h,'MaxStep' ,h);
disp('Simulación de un sistema lineal de segundo orden' )
[t, x]=ode45('sistema_segundo_orden_A' ,ts,cond_iniciales,opciones);
[n, m]=size(x(:,1));
y=zeros(n,m);
u=ones(n,m);
wn=1; rho=0.2;
for k=3:n %simulación del sistema discreto de segundo orden.
    y(k)=(1/(wn^2+(2*rho*wn)/h+1/h^2))*(wn^2*u(k)+(2/h^2+(2*rho*wn)/h)*y(k-1)-1/h^2*y(k-2));
end
figure
subplot(3,1,1); plot(t,x(:,1))
subplot(3,1,2); plot(t,y)
subplot(3,1,3); plot(t, x(:,1), t,y)
```

Robótica FCE BUAP

Código MATLAB 3: sistema_segundo_orden_A

Robótica, Período de primavera (24 de enero de 2020). Licenciatura en Ingeniería Mecatrónica.
Dr. Fernando Reyes Cortés. Facultad de Ciencias de la Electrónica, BUAP.

sistema_segundo_orden_A.m — MATLAB 2019a

```
function xp=sistema_segundo_orden_A(t,x)
    wn=1; % factor de amortiguamiento.
    rho=0.2; % freno mecánico o fricción (ganancia derivativa).
    A=[ 0, 1; % matriz A.
        -wn*wn, -2*rho*wn];
     %vector B.
    B=[0;
        wn*wn];
    u=1; %señal de entrada.
     %Sistema dinámico lineal.
    xp=A*x+B*u;
end
```

Robótica FCE
BUAP

Estimación de la derivada de la posición $x_1(t)$ por filtrado.

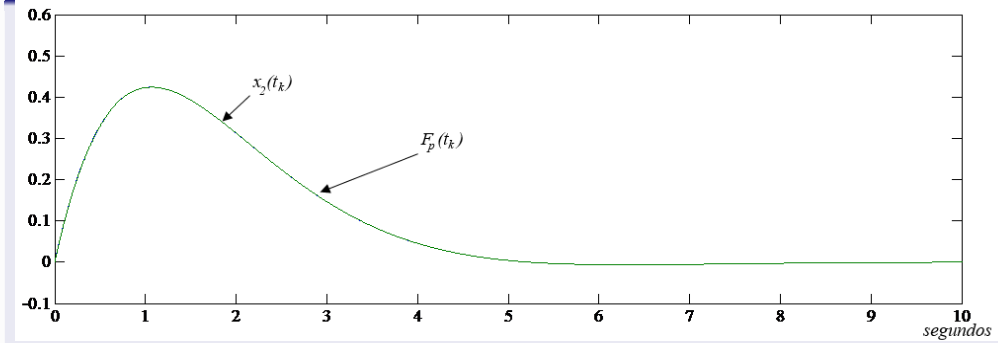


Figura 5: Estimación de la derivada de $x_1(t_k)$ por filtrado.

Algoritmo de diferenciación numérica (método de Euler)

Ejemplos de programación en **MATLAB** y aplicaciones prácticas con el método de Euler.

- Implementar en **MATLAB** el método de de Euler (diferenciación numérica).
- Calcular la derivada numérica de los siguientes casos de estudio $f(t) = \operatorname{sen}(t), t^2, 8t$ y $\cos(t)$:
  - $\operatorname{sen}(t)$.
  - t^2.
  - $8t$.
  - $\cos(t)$.

  considere $t \in [0,\ 10]$. Además el periodo de muestreo $h = 1$ mseg.
- Realizar una comparación técnica entre el método numérico de Euler con la solución analítica (presentar las gráficas en función del tiempo).

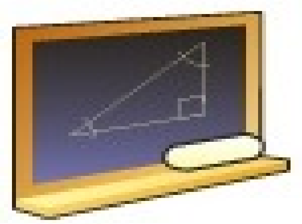


Figura 6: Ejemplos prácticos del algoritmo de diferenciación numérica.

Robótica FCE

BUAP