程式人

用十分鐘學會

《微積分、工程數學》及其應用

《電子電路電磁波、訊號語音影像處理》

陳鍾誠

2016 年 1 月 13 日

不管你念哪個科系

- 上了大學之後，幾乎都要念微積分

理工科系

- 學微積分好像還有點道理

但是法商學院、醫農學院

- 也都要上微積分

到底微積分是在學甚麼

- 又有甚麼用呢？

讓我們花幾分鐘

仔細思考一下

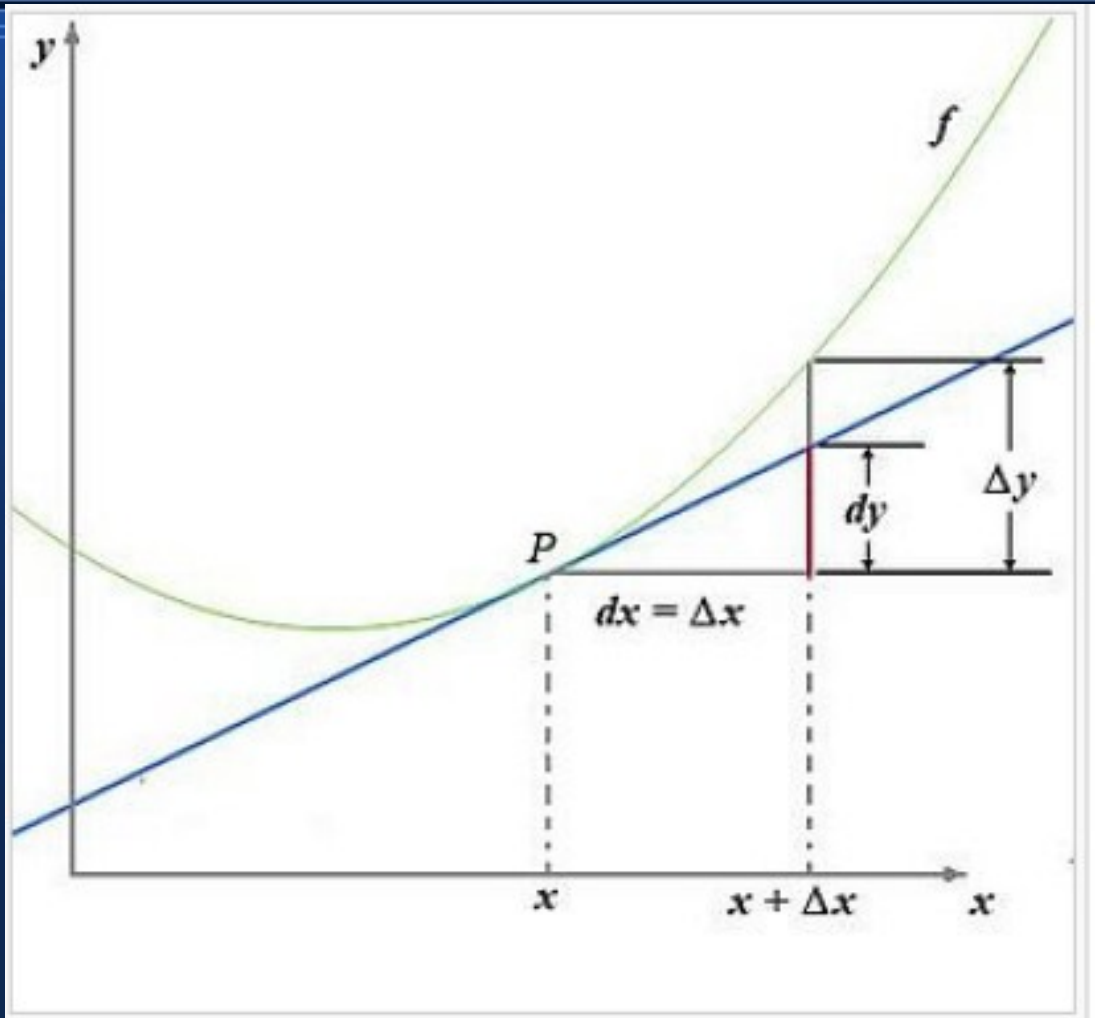
顧名思義

- 微積分所學的就兩件事
  - 微分
  - 積分

微分是在算斜率

- 也就是

dy/dx



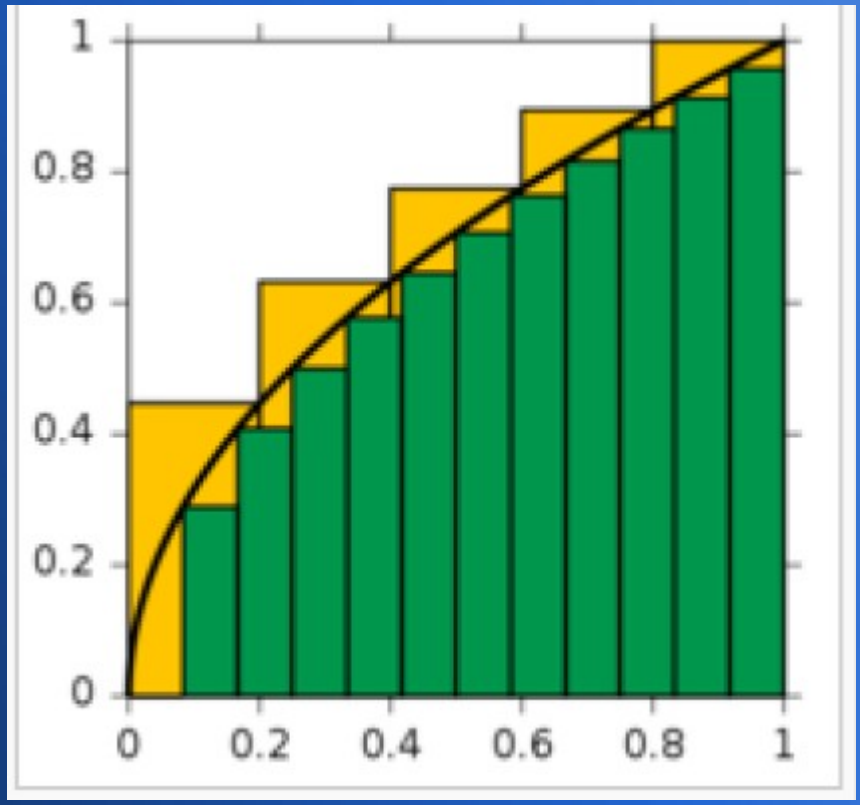
函數在一點的微分。其中紅線部分是微分量dy，而加上灰線部分後是實際的改變數Δy

積分是在算面積

- 也就是右圖的

  黑線下的部分

$$\int_a^b f(x)dx = \lim_{n\to\infty} \sum_{k=1}^{n} f(c_k)\Delta x_k$$



看完這兩張圖

- 差不多也就學完了

這樣的話

- 還要講一學期嗎？

其實、大部分的人學完微積分

- 還記得的東西，也就是上面那兩件事而已！

但是、你還記得極限嗎？

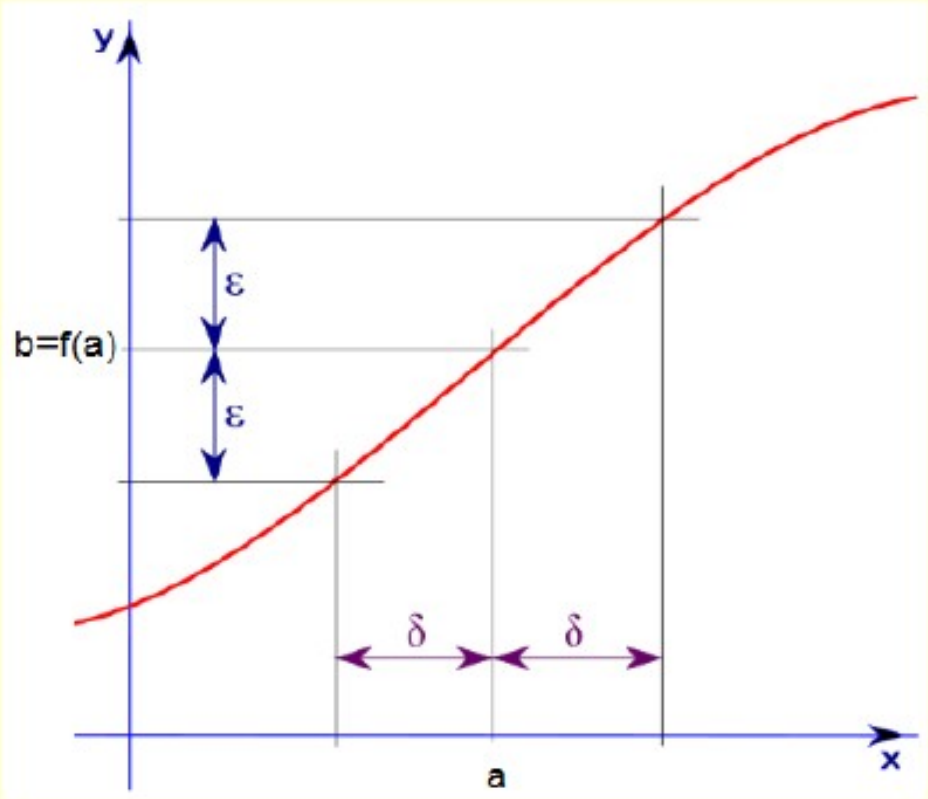
極限的定義

如果在 x 趨近於 a 時 f(x) 可以「任意接近」 b ，那我們就說 f(x) 趨近於 a 時的極限為 b，其數學符號定義如下。

$$\lim_{x \to a} f(x) = b$$

以上定義中的「任意接近」(arbitrarily close to) 的數學意義是：對於任何 $\epsilon > 0$，都存在一個 $\delta > 0$ 使得在 $0 < |x - a| < \delta$ 的情況下會滿足 $|f(x) - b| < \epsilon$ ，如下圖所示。

所以如果您想證明 f(x) 在 x=a 的極限存在，只要證明可以「任意接近」就行了。也就是找出滿足 $|f(x) - b| < \epsilon$ 的 δ 條件，並證明這個條件存在就行了。



那兩個希臘字母要怎麼唸呢？

- δ
- ε

你忘記了嗎？

大寫	小寫	寫法	念法
A	α	alpha	'æ lfə
B	β	beta	'betə
Γ	γ	gamma	'gæ mə
Δ	δ	delta	'dɛltə
E	ε	epsilon	'epsɪlən
Z	ζ	zeta	'zitə
H	η	eta	'eɪtə
Θ	θ	theta	'θitə
I	ι	iota	aɪ'ɔutə
K	κ	kappa	'kæ pə
Λ	λ	lambda	'læ pə
M	μ	mu	'mjuə

大寫	小寫	寫法	念法	提示
N	ν	nu	'nju	
Ξ	ξ	xi	'saɪ	cy, 為了與Ψψ 區分可念 k-si , 也有人念 zi
O	ο	omicron	'ɔʊmaɪkrən	
Π	π	pi	'paɪ	有 py 與 'paɪ 兩種念法
P	ρ	rho	'rɔʊ	
Σ	σ	sigma	'sɪgmə	
T	τ	tau	'tau	
Y	υ	upsilon	'jupsɪlən	與 E ε 的區分在第一個音為 u 或 a
Φ	φ	phi	'faɪ	fy
X	χ	chi	'kaɪ	ky
Ψ	ψ	psi	'psaɪ	cy, 為了與 Ξ ξ 區分可念 p-si, 也有人念 'psaɪ
Ω	ω	omega	'ɔʊmɛgə	

還有那些數學符號該怎麼唸呢？

符號	念法
$\hat{a}$	a hat
$\tilde{a}$	a tilde
$\bar{a}$	a bar
$\dot{a}$	a dot
f'	f prime
a_k	a sub k
a^k	a sup k
$\frac{a}{b}$	a over b
$\star$	star
$\wedge$	wedge
$\vee$	vee
$\forall$	forall

符號	說明	LaTex 寫法
$\simeq$	approximately equal to (趨近)	\simeq
$\equiv$	equivlent to (全等、定義)	\equiv
$\propto$	proportional to (正比)	\propto
∞	infinity (無限大)	\infty
$x \mapsto a$	x maps to a	x \mapsto a
$x \to a$	x approaches a	x \to a
$\lim x \to a f(x)$	f(x) — 當 x 趨近於 a 時，	\lim{x \to a} f(x)
$\arg\max_x f(x)$	最大化 f(x)	\arg \max_x f(x)
$\arg\min_x f(x)$	最小化 f(x)	\arg \min_x f(x)
$\lceil x \rceil$	ceil 函數 (天花板)	\lceil x \rceil
$\lfloor x \rfloor$	floor 函數 (地板)	\lfloor x \rfloor
$\hat{\theta}$	最大似然估計	\hat{\theta}

還是不會念嗎？

- Greek Alphabet（Rap 版）
  - http://www.youtube.com/watch?v=AvrDEegIo9g
- The (koine) Greek Alphabet Song
  - http://www.youtube.com/watch?v=3gaeIUsPJ-Y

讓我們請老師念給你聽！

好了、會念希臘字母之後

- 讓我們言歸正傳

對了、正傳到底是甚麼？

糟了、我忘了！

喔！對了、微積分中的符號！

- 那兩個符號代表非常小的數，通常會逼近到無窮小。
- 這種無窮小的概念稱為極限

問題是

- 到底學極限要做甚麼呢？

聽說、微積分是牛頓和萊布尼茲發明的！

艾薩克·牛頓爵士

艾薩克·牛頓爵士
戈弗雷·內勒作於1702年

出生	1643年1月4日（儒略曆1642年12月25日）[1] 英格蘭林肯郡埃爾斯索普村
逝世	1727年3月31日（儒略曆3月20日）[1]（84歲） 英格蘭倫敦肯辛頓
居住地	英格蘭
國籍	英格蘭
研究領域	神學、物理學、數學、天文學、自然哲學和鍊金術
任職於	劍橋大學、皇家學會

哥特佛萊德·威廉·萊布尼茲

萊布尼茲

西方哲學家 17世紀哲學家	
出生	1646年7月1日 德國萊比錫
逝世	1716年11月14日(70歲) 德國漢諾瓦
學派	理性主義
主要領域	數學、認識論、形上學、科學、神義論
著名思想	單子論

問題是

- 他們真的有發明極限概念嗎？
- 還是他們只發明了微分和積分？

如果沒有

- 那我們為甚麼要學極限概念？

極限概念可以幹嘛？

關於這些問題

- 都不屬於本文所討論的範圍

有問題的話

- 去問你的微積分老師

在本文中

- 我們只討論
  - 微積分怎麼用？
  - 還有用在哪裡？

對於程式人而言

- 學完微積分，一定要會
  - 《微分》和《積分》的程式

這應該不難

dx = 0.001

```
function df(f, x) {
  var dy = f(x+dx)-f(x);
  return dy/dx;
}
```

微分函數

```
function integral(f, a, b) {
  var sum=0.0, x;
  for (x = a; x <=b; x+=dx) {
    sum += f(x)*dx;
  }
  return sum;
}
```

積分函數

好了，寫完了！

- 接下來還可以幹嘛？

程式人

- … 我想不出來了！

好吧！

- 既然這樣
- 那就回家找媽媽
- 然後洗洗睡了！

喂！等一下

- 你是說我學了一學期
- 就只為了寫這兩個程式嗎？

阿不然哩！

- 你還希望微積分有甚麼用途嗎？
- 不就是計算斜率和面積嗎？
- 上面那兩個函數就做完了阿！

這 …

- 程式人心想：雪特雪特雪特 . . .

好像有點不太對勁！

程式人問

- 那不是還有工程數學嗎？
- 裡面不是有個《微分方程》嗎？
- 那對寫程式沒用嗎？

微積分老師

- ...

程式人沒得到解答

- 跑去問教程式的老師

教程式的老師

- 你是問微積分和工程數學的用途嗎？
- 那對寫程式確實沒什麼用

但是

- 好像對電子電路有點用
- 你要不要去問問電子系的老師

程式人找到電子系老師

- 請問微分方程有甚麼用？

電子系老師說

- 微分方程可以用在解
  - 有電感或電容的電路上

像是這個

6.4.1. RC 電路 (一階)

RC 串聯電路的暫態響應 (充放電時的反應)

一個最簡單的 RC 電路是由一個電阻器和一個電容器串聯組成的，如右圖所示。

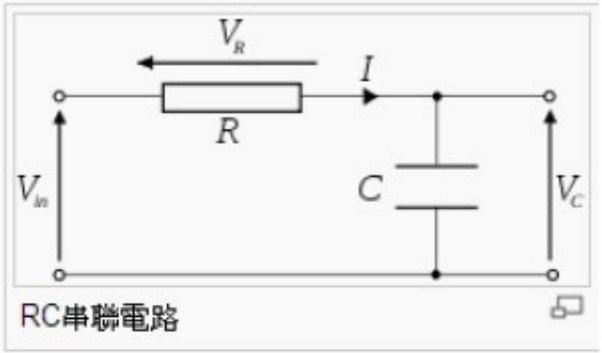


插圖 5: RC 電路的示意圖

當我們外加直流電源 V 伏特對 RC 電路的電容充電時，由於電子無法穿過電容，因此流過電阻的電流全數流入了電容，因此可得微分方程式

$$I = C\frac{dV_C}{dt} = \frac{V_{in} - V_C}{R}$$

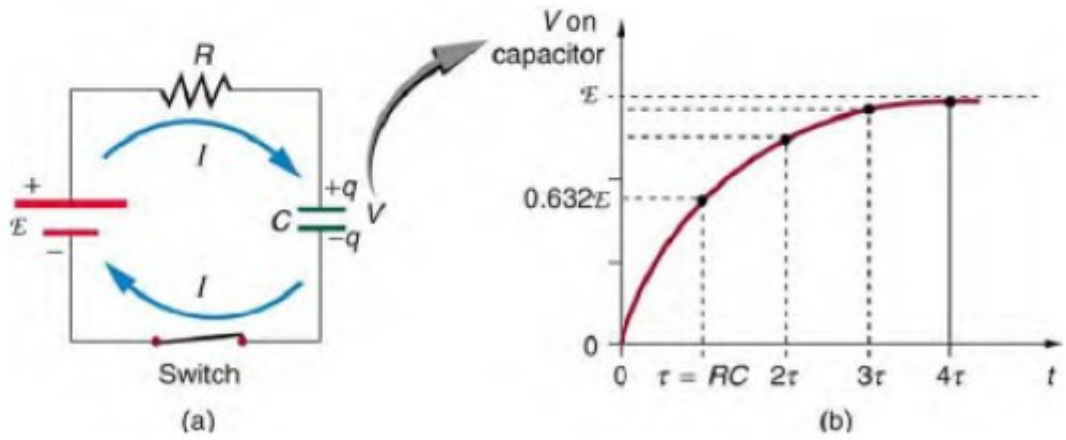


插圖 6: RC 電路對電容充電時的情況

還有這個

同樣的，當我們將外加直流電源拿掉，變成放電的時候，電容內的電子流出後全數都會通過電阻釋放回去，所以可得微分方程式

$$C\frac{dV_C}{dt} = I = -\frac{V_C}{R}$$ [19]

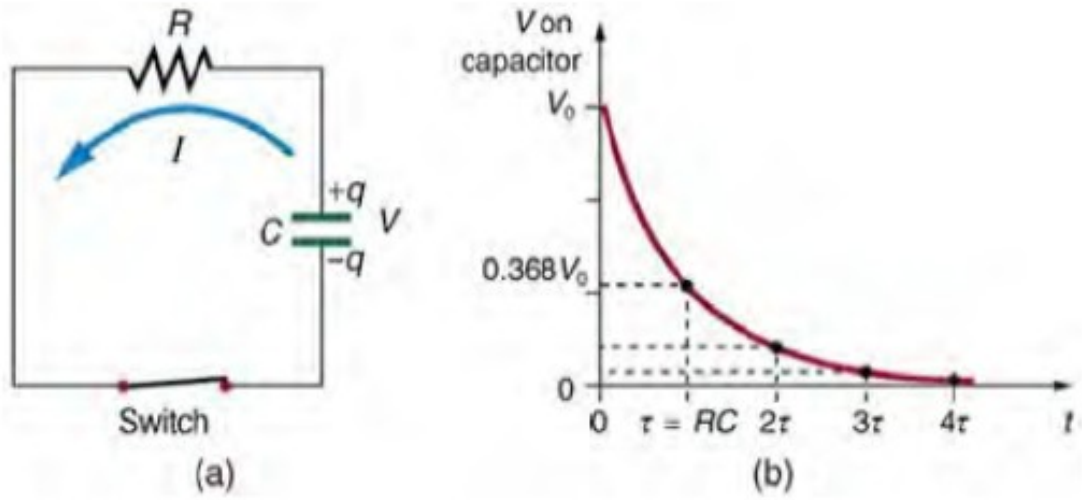


插圖 7: RC 電路電容放電時的情況

---

19 說明：充電開始時電壓 $V_C = 0; V_{in} >> V_C$，而放電時 $V_{in} = 0; V_C >> V_{in}$，以上兩種情形，由於克希荷夫定律，流過電阻的電流全數流入了電容，回流時也全數從電容流回電阻，因此其微分方程式都是如下形式，只是初始條件不同 (充電時$K = V_{in}$，放電時 $K = 0$)，所以解出來的答案也不同罷了。

$$C\frac{dV_C}{dt} = \frac{K' - V_C}{R} = \frac{-V_C}{R} + K$$

程式人

- 就這樣嗎？

電路學老師

- 等等，那只是 RC 電路
- 還有 RL, LC, RLC 電路沒講

RL 電路

脈衝響應 [編輯]

每一種電壓的衝激響應是對應傳輸函數的反拉普拉斯變換。它代表電路對於包含脈衝或狄拉克δ函數的輸入電壓的響應。

電感元件電壓的響應為：

$$h_L(t) = \delta(t) - \frac{R}{L}e^{-tR/L}u(t) = \delta(t) - \frac{1}{\tau}e^{-t/\tau}u(t)$$

這裡$u(t)$是單位階躍函數且

$\tau = \frac{L}{R}$為時間常數。

類似的，電阻器電壓的響應為：

$$h_R(t) = \frac{R}{L}e^{-tR/L}u(t) = \frac{1}{\tau}e^{-t/\tau}u(t)$$

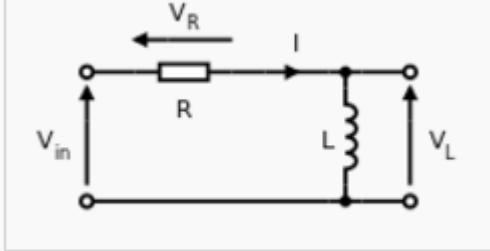
零輸入響應 [編輯]

RL電路的零輸入響應（Zero input response， ZIR）描述了電路在不連接輸入信號源的情況下、達到穩定電壓和電流時的工作狀態。[4]因為它沒有外接輸入信號，因此得名。

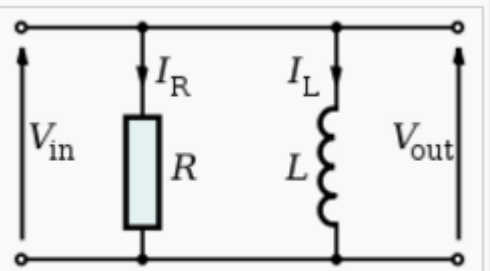
一個RL電路的零輸入響應為：

$$i(t) = i(0)e^{-(R/L)t} = i(0)e^{-t/\tau}.$$

其中τ是時間常數。



RL電路的串聯形式



RL電路的並聯形式

RC 電路

暫態響應 [編輯]

一個最簡單的RC電路是由一個電阻器和一個電容器串聯組成，而當一個電路只由一個充電的電容器和一個電阻器組成時，電容器會釋出電流給電阻器。 根據克希荷夫電路定律我們可求得電流於電容器所耗時間內產生的變化，其結果經由 線性微分方程

$$C\frac{dV}{dt} + \frac{V}{R} = 0$$

解求得時，其結果於指數衰變函數

$$V(t) = V_0 e^{-\frac{t}{RC}}$$

RC並聯電路 [編輯]

電路電流

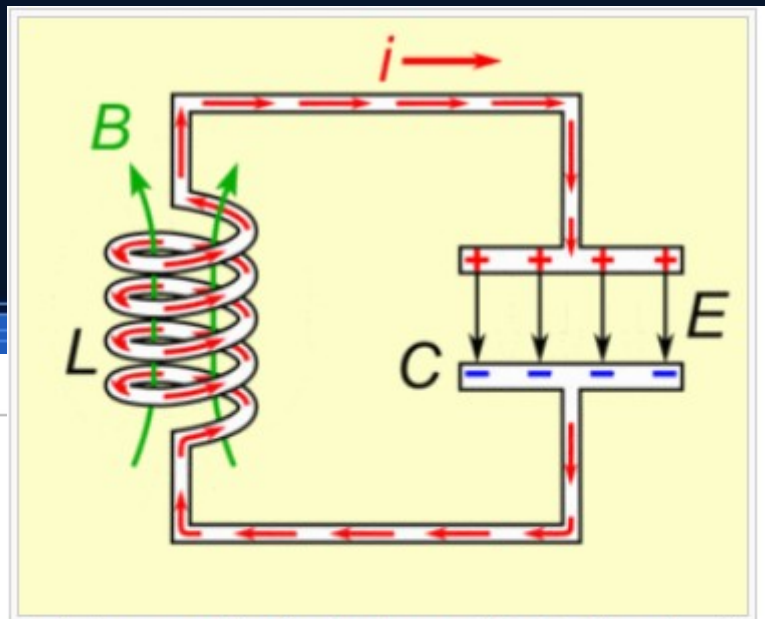
$$I = I_R + I_C$$

$$I_R = \frac{V_{in}}{R}$$

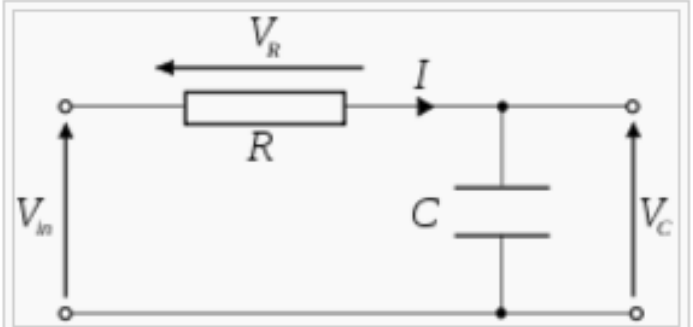
$$I_C = j\omega C V_{in} = C\frac{dV_{in}}{dt}$$

阻抗

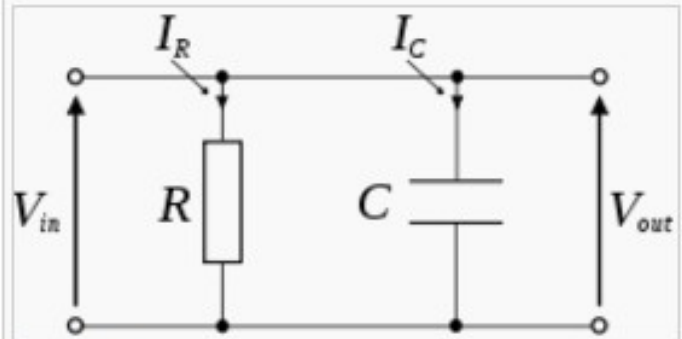
$$Z = \frac{V_{out}}{I_{in}} = \frac{R}{1 + sRC}$$



動畫演示了LC電路（無電阻的RLC電路）的工作。電荷在電容器極板和電感之間來回傳遞。能量在電容器的電場 (E) 和電感的磁場 (B) 之間來回振蕩。 RLC電路工作情況類似，不同之處在於，由於電路中的電阻，隨著時間變化振蕩電流衰減至零。



RC串聯電路



RC並聯電路

RLC 串聯電路

串聯RLC電路：電阻、電感和電容

RLC串聯電路 [編輯]

在此電路中，三個元件均與電壓以串聯方式連接。其主要的微分方程可將三個元件的本構方程代入基爾霍夫電壓定律（KVL）獲得。由基爾霍夫電壓定律：

$$v_R + v_L + v_C = v(t)$$

其中v_R, v_L, v_C分別為R、L、C兩端的電壓，$v(t)$為隨時間變化的電源的電壓。將本構方程代入得到：

$$Ri(t) + L\frac{di}{dt} + \frac{1}{C}\int_{-\infty}^{\tau=t} i(\tau)\,d\tau = v(t)$$

在電源電壓為常數的情況下，對上式求導，並且除以L，得到以下二階微分方程：

$$\frac{d^2i(t)}{dt^2} + \frac{R}{L}\frac{di(t)}{dt} + \frac{1}{LC}i(t) = 0$$

此方程可以寫成更常用的形式：

$$\frac{d^2i(t)}{dt^2} + 2\alpha\frac{di(t)}{dt} + {\omega_0}^2 i(t) = 0$$

α稱為「衰減量」，用於衡量當移除外部輸入後，此電路的瞬態響應衰減的速率。ω_0為角共振頻率。[1]此二係數由下式給出：[2]

$$\alpha = \frac{R}{2L}\ ,\ \omega_0 = \frac{1}{\sqrt{LC}}$$

阻尼係數ζ是另一個常用的參數，定義為α與ω_0的比值：

$$\zeta = \frac{\alpha}{\omega_0}$$

阻尼係數也可以由R、L、C求得：

$$\zeta = \frac{R}{2}\sqrt{\frac{C}{L}}$$

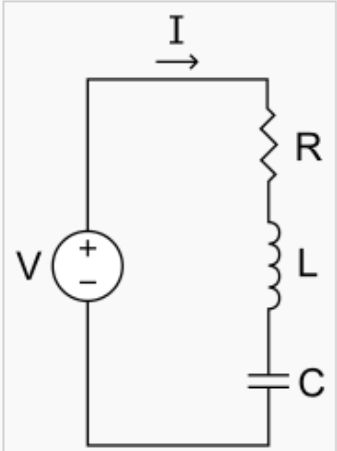


圖 1. RLC串聯電路
V - 電源電壓
I - 電路電流
R - 電阻
L - 電感
C - 電容

RLC 並聯電路

RLC並聯電路 [編輯]

RLC並聯電路的特性可以利用電路的對偶性，將RLC並聯電路視為RLC串聯電路的對偶阻抗來處理，就可以用類似RLC串聯電路的分析方式來分析RLC並聯電路。

RLC並聯電路的衰減量α可以用下式求得[11]：

$$\alpha = \frac{1}{2RC}$$

而其阻尼係數為：

$$\zeta = \frac{1}{2R}\sqrt{\frac{L}{C}}$$

若不考慮$1/2$的係數，RLC並聯電路的阻尼係數恰好是RLC串聯電路阻尼係數的倒數。

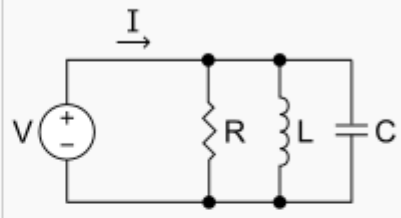


圖 5. RLC 並聯電路
V - 電源電壓
I - 電路電流
R - 電阻
L - 電感
C - 電容

頻域 [編輯]

將並聯各元件的導納相加，即為此電路的導納：

$$\frac{1}{Z} = \frac{1}{Z_L} + \frac{1}{Z_C} + \frac{1}{Z_R} = \frac{1}{j\omega L} + j\omega C + \frac{1}{R}$$

電容、電阻及電感並聯後，在共振頻率的阻抗為最大值，和電容、電阻及電感串聯的情形恰好相反，RLC並聯電路是抗共振電路（antiresonator）。

右圖中可以看出若用定電壓驅動時，電流的頻率響應在共振頻率$\omega_0 = \frac{1}{\sqrt{LC}}$處有最小值。若用定電流驅動，電壓的頻率響應在共振頻率處有最大值，和RLC串聯電路中，電流的頻率響應圖形類似。

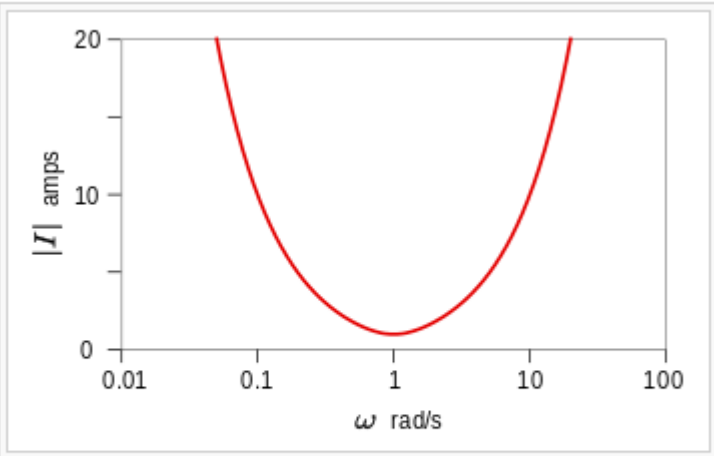


圖 6. *正弦響應分析*
以R = 1 歐姆、C = 1 法拉、L = 1 亨利、V = 1.0 伏特來進行正規化

程式人

- 我看不懂
- 那裏面的 i, j 是甚麼？

電路學老師

- 那些 i, j 是虛數

定義是：

$$i^2 = -1$$

或者

$$i = +\sqrt{-1}$$

程式人

- 那虛數有甚麼用？

電路學老師

- 交流電的波形可以用三角函數表示
- 實數加虛數所形成的複數，很適合用來描述電波

程式人

- 不懂
- 可以解釋一下嗎？

關於這個問題

- 我們得先回到《尤拉公式》上，才有辦法說明！
- 不過在說明尤拉公式前，必須先說明《尤拉數》與《尤拉函數》。

尤拉數 e 與尤拉函數 e^x

尤拉數	公式	說明
定義	$e = \lim_{n\to\infty}(1+\frac{1}{n})^n$	尤拉數是一種逼近值
特性	$\int_1^e \frac{1}{x}dx = 1$	1/x 從 1 到 e 的積分為 1
泰勒級數	$e = \lim_{n\to\infty} 1+\frac{1}{1!}+\frac{1}{2!}+\ldots+\frac{1}{n!}$	尤拉數的泰勒級數非常簡單
微分	$\frac{d}{dx}e^x = e^x$	尤拉函數是微分運算的單位元素
積分	$\int e^x dx = e^x$	尤拉函數是積分運算的單位元素

然後、是尤拉公式

尤拉公式 (Euler's Formula)

$$e^{ix} = cos(x) + i * sin(x)$$

程式人

- 我看不懂 …
- 為何 $e^{ix} = cos(x) + i * sin(x)$ 呢？

喔！你還記得泰勒展開式嗎？

程式人

- 我 … 忘了！

以下是 f(x) 在 a 點的泰勒展開式

泰勒展開式

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \ldots + \frac{f^k(a)}{k!}(x-a)^k + \ldots = \sum_{k=0}^{\infty} \frac{f^k(a)}{k!}(x-a)^k$$

如果 a=0 ，寫起來比較簡單

$$f(x) = f(0) + \frac{f'(0)}{1!}x + \ldots + \frac{f^k(0)}{k!}x^k + \ldots = \sum_{k=0}^{\infty} \frac{f^k(0)}{k!}x^k$$

零點的泰勒展開式又稱為麥克羅林級數

如果你對尤拉公式的兩邊
各取泰勒展開式，就會得到

$$e^{ix} = 1 + i\frac{x}{1!} - \frac{x^2}{2!} - i\frac{x^3}{3!} + \ldots$$

$$= \quad cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} + \ldots$$

$$+ i \; sin(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} + \ldots$$

於是

- 看來原本沒關係的兩個式子
- 透過泰勒展開式竟然對上了

其實

- e^{ix} 中的 i 非常的重要
- 請您稍微想想

在極座標的體系裏面

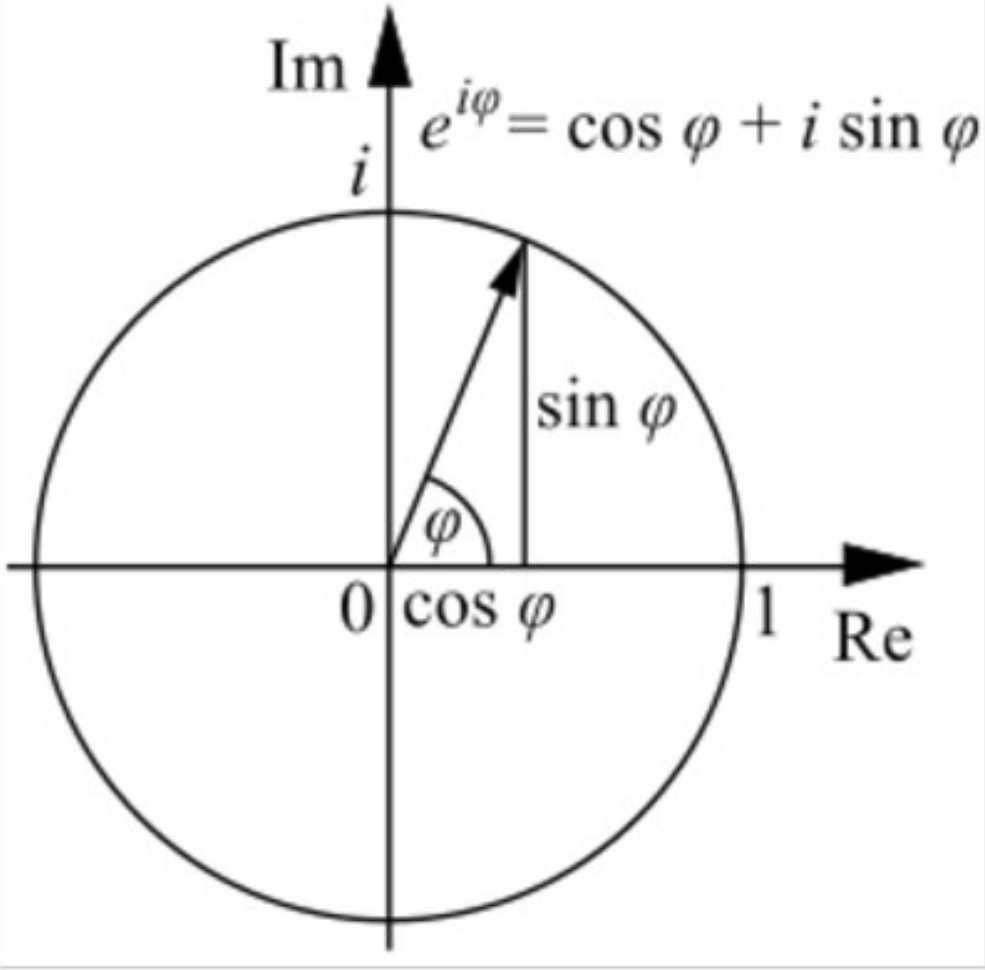
- e^{ix} 和 $e^{i(x+\varphi)}$ 之間，

  代表轉了 φ 的角度

- 但這個轉動對 $f(x)=e^{ix}$

  卻變成了移動，從 f(x)

  移到了 f(x+φ)



An illustration of a complex number plotted on the complex plane using Euler's formula

轉動是非線性的，很難描述

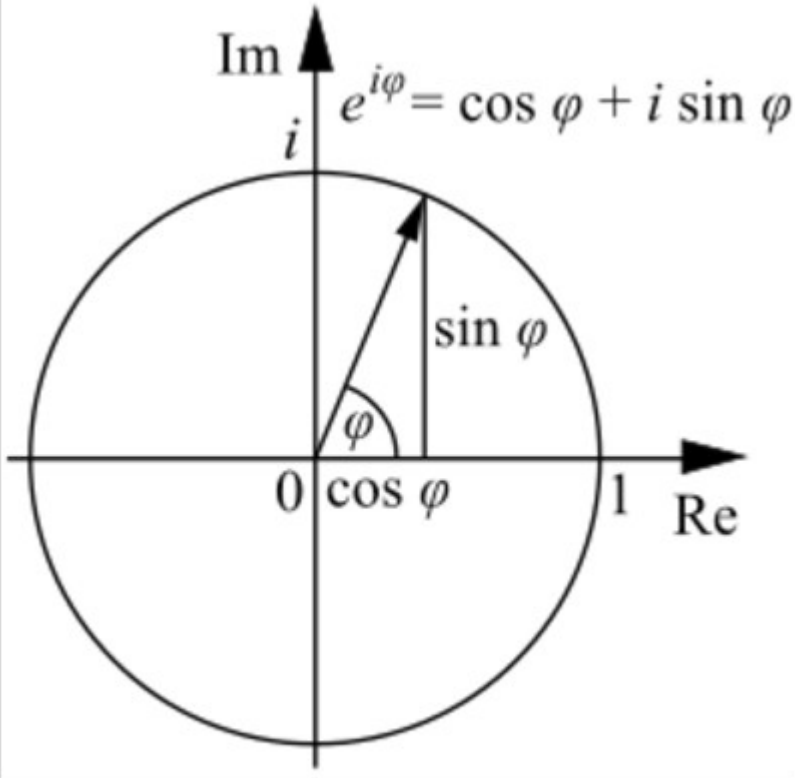
- 但移動是線性的，很好描述
- 能將非線性的轉動變成線性的移動，會讓事情簡單很多。

而且當我們把尤拉公式放進來之後

- 就會發現下列情況
  - $f(x)=e^{ix}=\cos(x+\varphi)+i\sin(x+\varphi)$
  - $f(x+\varphi)=e^{i(x+\varphi)}=\cos(x+\varphi)+i\sin(x+\varphi)$
- 於是轉動變成了 e^{ix} 的移動，又變成了波的位移

稍微整理一下，會發現

- 三角函數本來就是從圓的概念來的，波動與轉動本來就是一體的兩面，所以尤拉複函數 e^{ix} 的移動既然代表了極座標的轉動，那自然就能用來描述波動了。
- 只是我們透過微積分的泰勒展開式繞了一圈又回來了而已。



An illustration of a complex number plotted on the complex plane using Euler's formula

但是

- 這樣大費周章繞了一圈，到底有甚麼價值呢？

這就是

- 數學神奇的地方了！

尤拉的發現

- 被傅立葉拿來逼近函數
- 結果形成了傅立葉轉換

到底

- 傅立葉轉換是甚麼呢？

其實不太難

- 只是把不同週期的尤拉複函數加起來而已！

但是在直覺意義上

- 其實是
  - 用三角函數的組合
  - 去逼近任何一個複函數

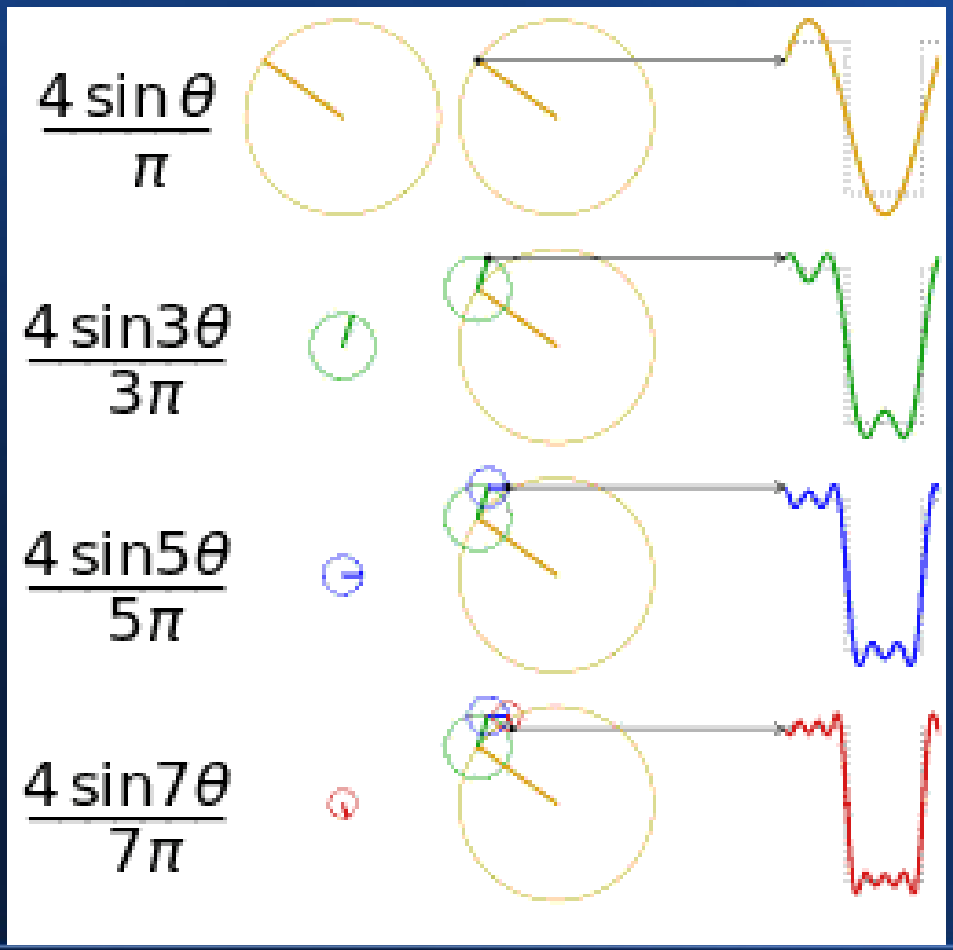
怎麼說呢？

- 讓我們看看下列函數

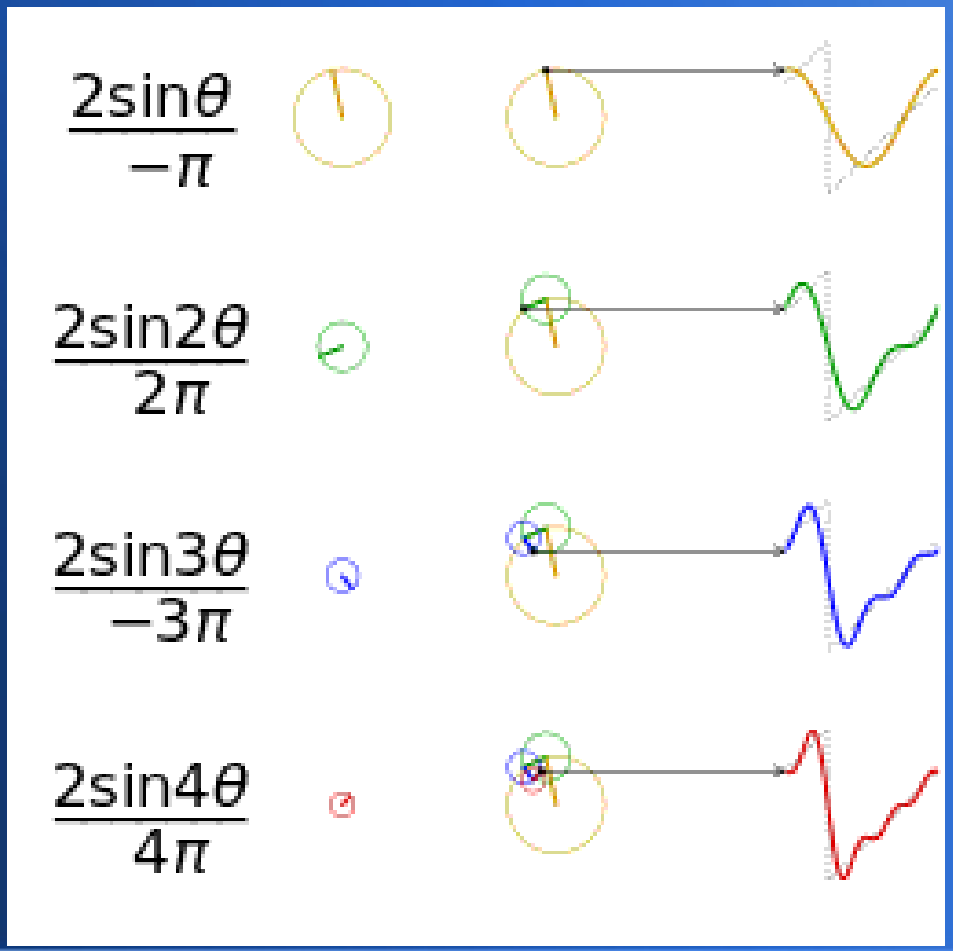
$$f(x) = \sum_{n=0}^{\infty} a_n cos(nx) + b_n sin(nx) i$$

如果我們把數個不同週期的 sin 函數加總會得到類似下圖的結果

來源 :https://en.wikipedia.org/wiki/File:Fourier_series_square_wave_circles_animation.gif
動畫 :https://en.wikipedia.org/wiki/File:Fourier_series_sawtooth_wave_circles_animation.gif



上列組合很接近方波



上列組合很接近三角波

數學上可以證明

- 傅立葉級數可以用來逼近任何一個週期複函數。

然後、當我們將傅立葉級數的加總式

- 改寫成積分式的時候，就變成了傅立葉《逆轉換》

$$f(x) = \sum_{n=0}^{\infty} a_n cos(nx) + b_n sin(nx)$$

$$= \sum_{n=0}^{\infty} F_n e^{inx}$$

$$\rightarrow \int_{-\infty}^{\infty} F(n) e^{inx} dn$$

以下是傅立葉逆轉換的各個部分

	正轉換	反轉換（逆向轉換）
傅立葉轉換	?	$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(n) e^{inx} dn$
實部	?	$f(x).r = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(n) cos(nx) dn$
虛部	?	$f(x).i = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(n) sin(nx) dn$

有逆轉換當然有正轉換

傅立葉轉換	$F(n) = \int_{-\infty}^{\infty} f(x)e^{-inx}dx$	$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(n)e^{inx}dn$

有連續形式當然也可以改成離散形式

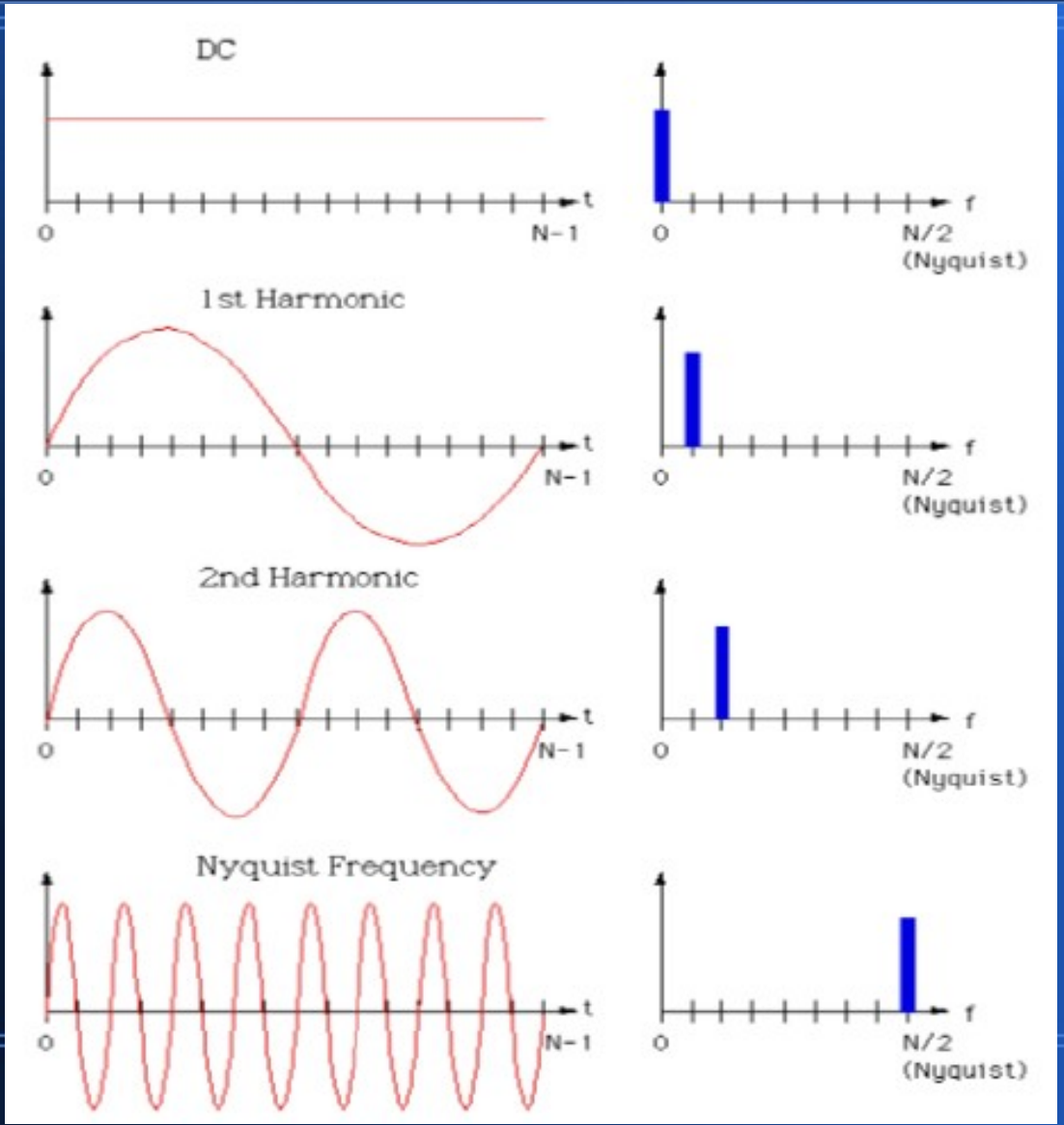
	正轉換	反轉換（逆向轉換）
離散傅立葉轉換	$F(n) = \sum_{x=0}^{N-1} f(x) e^{-i2\pi \frac{x}{N} n}$	$f(x) = \frac{1}{N} \sum_{n=0}^{N-1} F(t) e^{i2\pi \frac{n}{N} x}$

然後你就可以利用傅立葉轉換

- 將任何的波型轉換到另一邊的《頻譜領域》
- 原本的 sin, cos 波動在《頻譜領域》就只剩下幾個些點

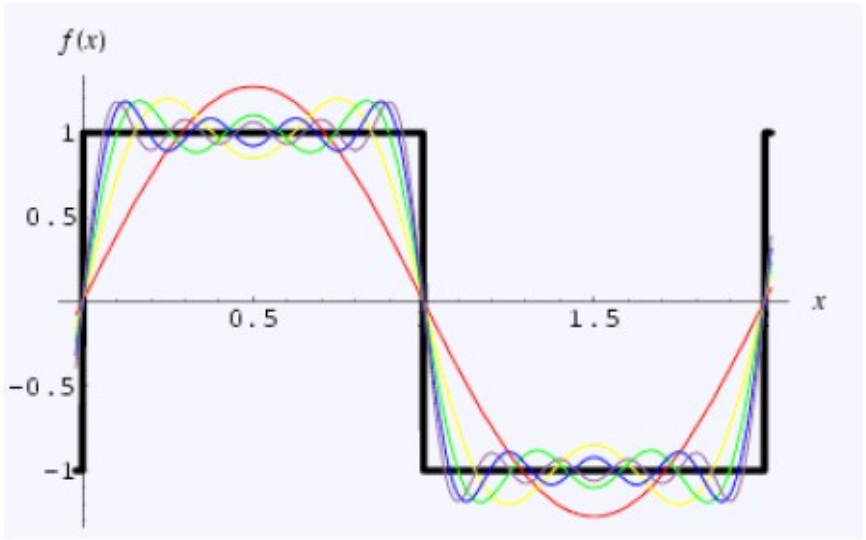
像是單純的 sin 波轉過去就只剩一個點

圖片來源： http://paulbourke.net/miscellaneous/dft/



複合波其實很可能用幾個系數就能代表了

以下圖形代表利用 sin(nx) 的波形去組合出函數 f(x) 的一個情況，其中：



紅色線代表 $c_1 sin(1x)$

黃色線代表 $c_1 sin(1x) + c_2 sin(2x)$

綠色線代表 $c_1 sin(1x) + c_2 sin(2x) + c_3 sin(3x)$

藍色線代表 $c_1 sin(1x) + c_2 sin(2x) + c_3 sin(3x) + c_4 sin(4x)$

紫色線代表 $c_1 sin(1x) + c_2 sin(2x) + c_3 sin(3x) + c_4 sin(4x) + c_5 sin(5x)$

您可以看到

- 在《頻譜領域》，係數 sin(a x) 中 a 係數越小的表頻率越低
- 像是 sin(2x) 的頻率是 sin(1x) 的兩倍
  - 由於週期是頻率的倒數，所以 sin(2x) 的週期是 sin(1x) 的一半。

而且、我們可以寫程式來做傅立葉轉換

- 只要按照下列算式實作就行了

	正轉換
離散傅立葉轉換	$F(n) = \sum_{x=0}^{N-1} f(x) e^{-i2\pi \frac{x}{N} n}$
實部	$F(n).r = \sum_{x=0}^{N-1} f(x).r * cos(-2\pi \frac{x}{N} n) - f(x).i * sin(-2\pi \frac{x}{N} n)$
虛部	$F(n).i = \sum_{x=0}^{N-1} f(x).r * sin(-2\pi \frac{x}{N} n) + f(x).i * cos(-2\pi \frac{x}{N} n)$

當然也可以寫出逆轉換

	正轉換
離散傅立葉轉換	$F(n) = \sum_{x=0}^{N-1} f(x) e^{-i2\pi \frac{x}{N} n}$
實部	$F(n).r = \sum_{x=0}^{N-1} f(x).r * cos(-2\pi \frac{x}{N} n) - f(x).i * sin(-2\pi \frac{x}{N} n)$
虛部	$F(n).i = \sum_{x=0}^{N-1} f(x).r * sin(-2\pi \frac{x}{N} n) + f(x).i * cos(-2\pi \frac{x}{N} n)$

以下是我所寫的慢速傅立葉轉換

```
// 慢速傅立葉轉換
void SFT(Complex f[], int N, Complex F[]) {
        int n, x;
        double Nf = 1.0;
        for (n=0; n<N; n++) { F[n].r = 0.0; F[n].i = 0.0; }
        for (x=0; x<N; x++) {
            for (n=0; n<N; n++) {
                Complex exp  = cexpi((-2.0*M_PI*x)/N*n);
                Complex fexp = cmul(f[x], exp);
                  F[n] = cadd(F[n], fexp);
            }
        }
```

還有慢速傅立葉逆轉換

```
// 慢速傅立葉反轉換
void iSFT(Complex F[], int N, Complex f[]) {
        int n, x;
        double Nf = 1.0/N;
        for (x=0; x<N; x++) { f[x].r = 0.0; f[x].i = 0.0; }
        for (n=0; n<N; n++) {
                for (x=0; x<N; x++) {
                        Complex exp = cexpi((2.0*M_PI*x)/N*n);
                        Complex Fexp = cmul(F[n], exp);
                        Fexp.r /= N; Fexp.i /= N;
                        f[x] = cadd(f[x], Fexp);
                }
        }
}
```

當然、演算法課程會告訴你

- 那個太慢了，還可以更快
- 當然程式碼也會變長...
- 這就是快速傅立葉轉換 FFT

程式人

- 問題是：這些轉換到底有甚麼用呢？

傅立葉

- 如果我們用取樣的方式記錄一個 sin 波每秒紀錄一千點，那一分鐘就要 60K
- 但若用傅立葉轉換轉到《頻譜領域》，竟然只剩下一個點。
- 這樣我們就不用記那 60K 個點，只要記一個振幅系數就好了

於是

- 我們就得到一個超級程式
- 壓縮率達到 60K 倍，也就是六萬倍。

當然

- 現實世界的波形沒有那麼完美，像是語音、電波或圖片，通常都不是單純的 sin 波或 cos 波。
- 但是人的眼睛和耳朵的分辨率通常有限，特別是眼睛。

眼睛具有自動過濾功能

- 通常你看一張有樹的圖片，不會看到每一片葉子，而是看到整棵樹。
- 所以把頻率很高的部分過濾掉，其實對人的眼睛而言，不太會有感覺。
- 於是我們就只要留下低頻的部分，將高頻部分濾除或用很少位元表示。

這種濾除高頻的方式

- 正是 JPEG 影像檔所採用的方式
- 如果您將同一張圖分別存成 JPG 和 BMP 兩種不同格式，會發現 BMP 通常比 JPG 大上十幾二十倍。
- 這就是傅立葉轉換的威力了。（事實上 JPEG 只用到了實數部分，所以是 cos 轉換）。

同樣的、在訊號和語音處理領域

- 傅立葉轉換也能發揮很好的效果
- 於是微積分對連續函數的數學研究，竟然成了《訊號、語音、影像》處理上的利器。

這正是數學神奇的地方

- 也是《電子資訊》領域為何要學《微積分和工程數學》這些課程的原因。

希望

- 這份投影片有解答到您對《為何要學微積分》的疑惑。

下次、當您無法理解

- 為何要學那些數學的時候

或許

- 就不會那麼排斥那些符號

西洋文明

- 正是在一種不求《有用》的心態之下
- 反而走出了一條康莊大道

而數學

- 正是西洋文明當中，最令人激賞的那個璀璨寶石。

物理和數學

- 可以說是西洋科學的兩大支柱
- 在這樣的基礎上，發展出各式各樣的神奇科技

希望

- 我們的十分鐘系列

能夠重新燃起

- 您對這些學問的 . . .

渴望！