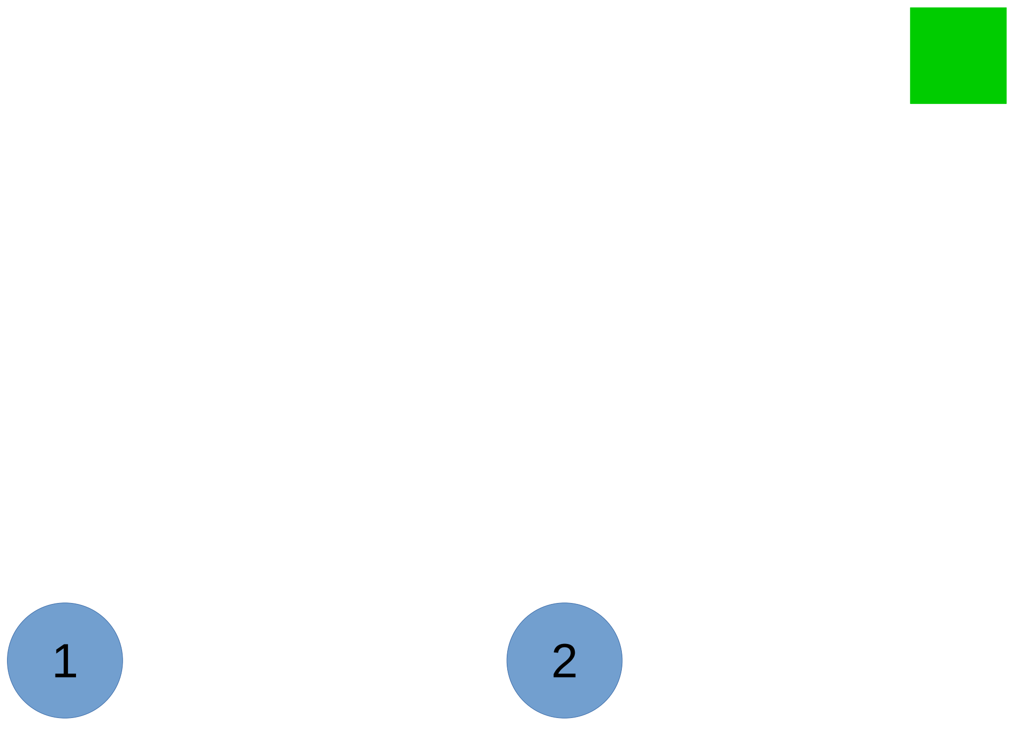
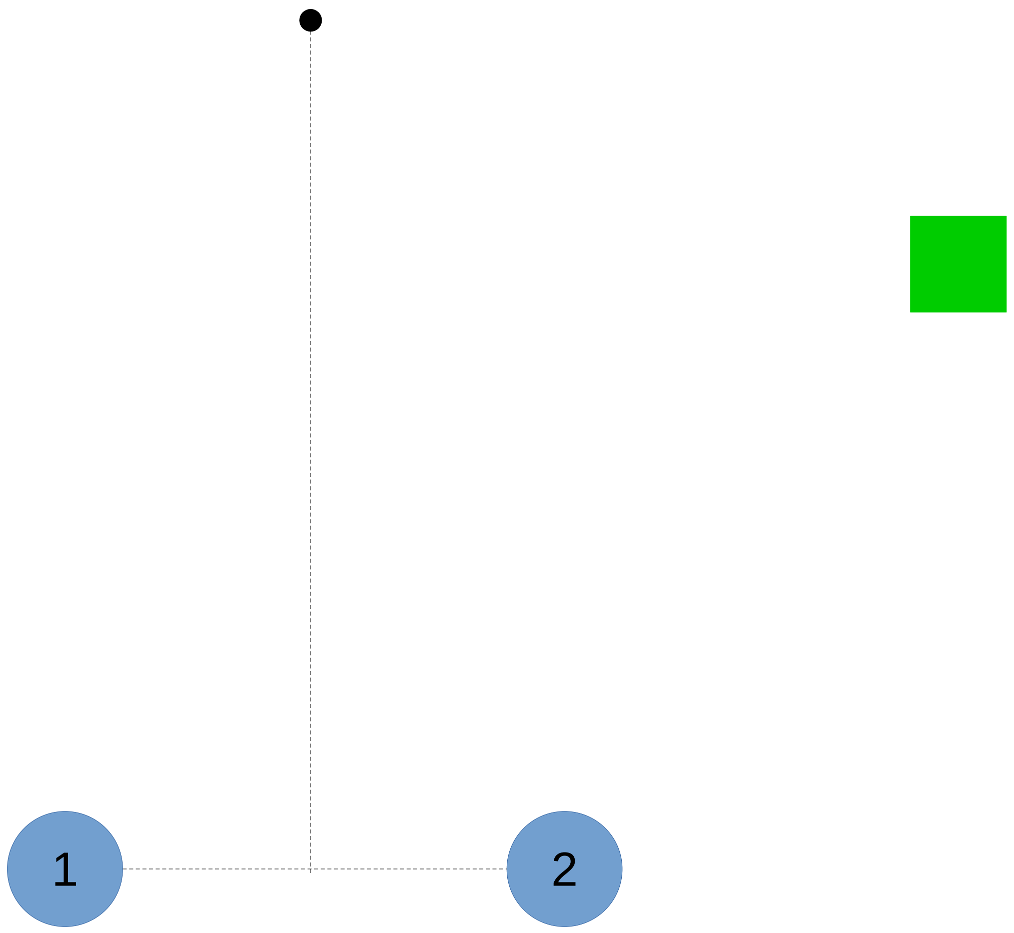
Estimación de Dirección de Arribo de **una** Fuente Sonora

Arreglo Simple de Micrófonos

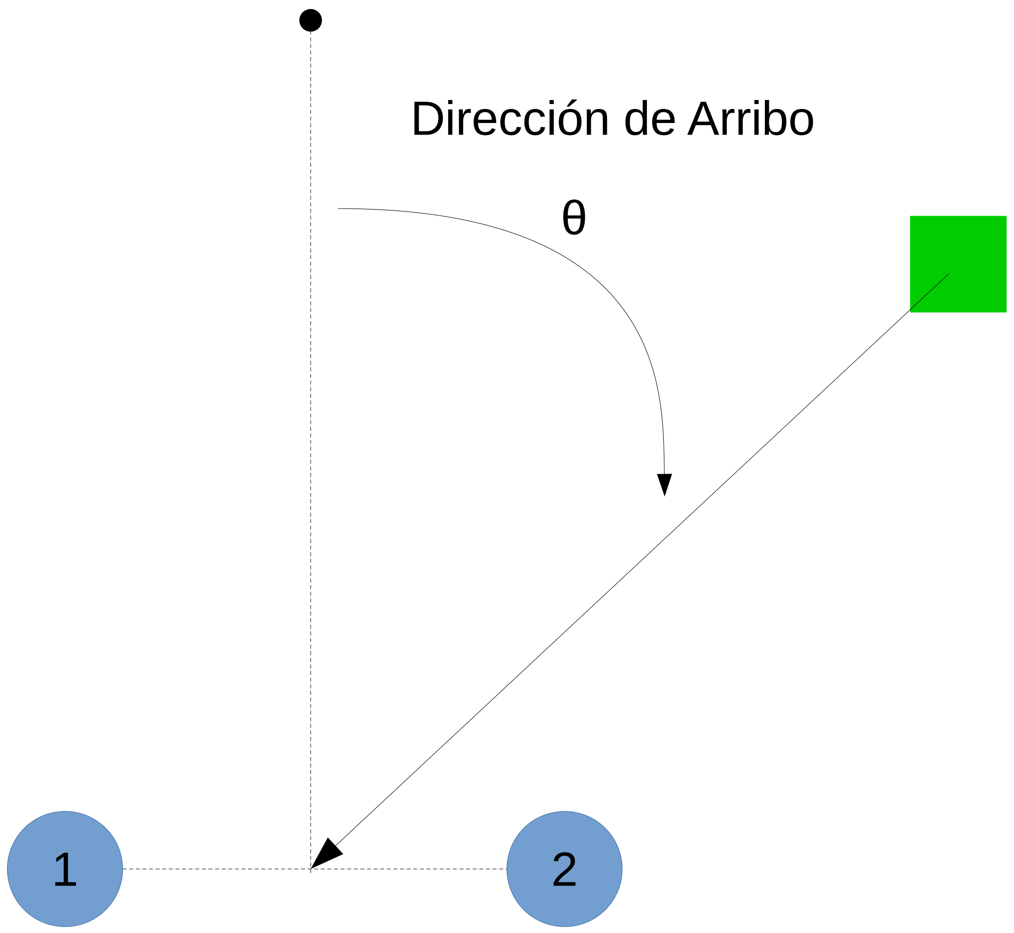
Arreglo Simple de Micrófonos



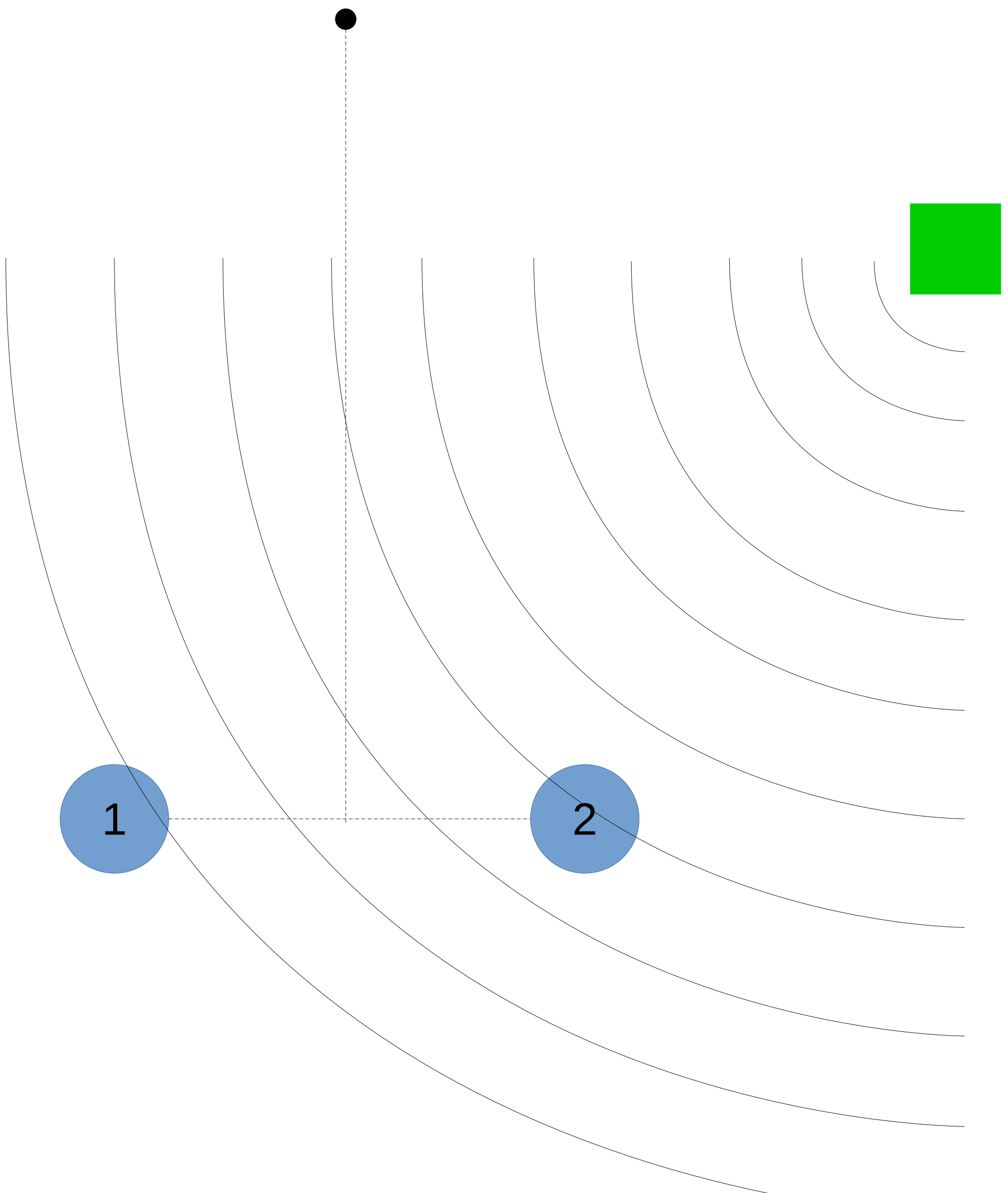
Arreglo Simple de Micrófonos



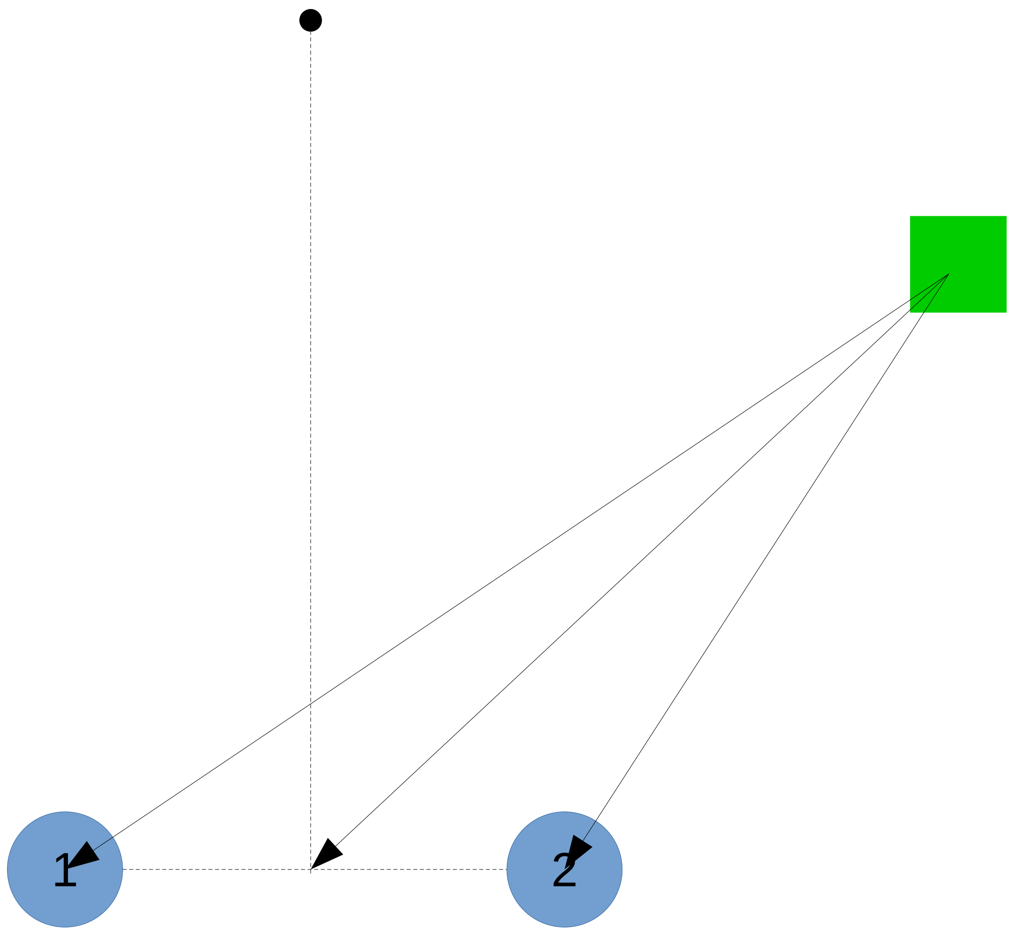
Arreglo Simple de Micrófonos



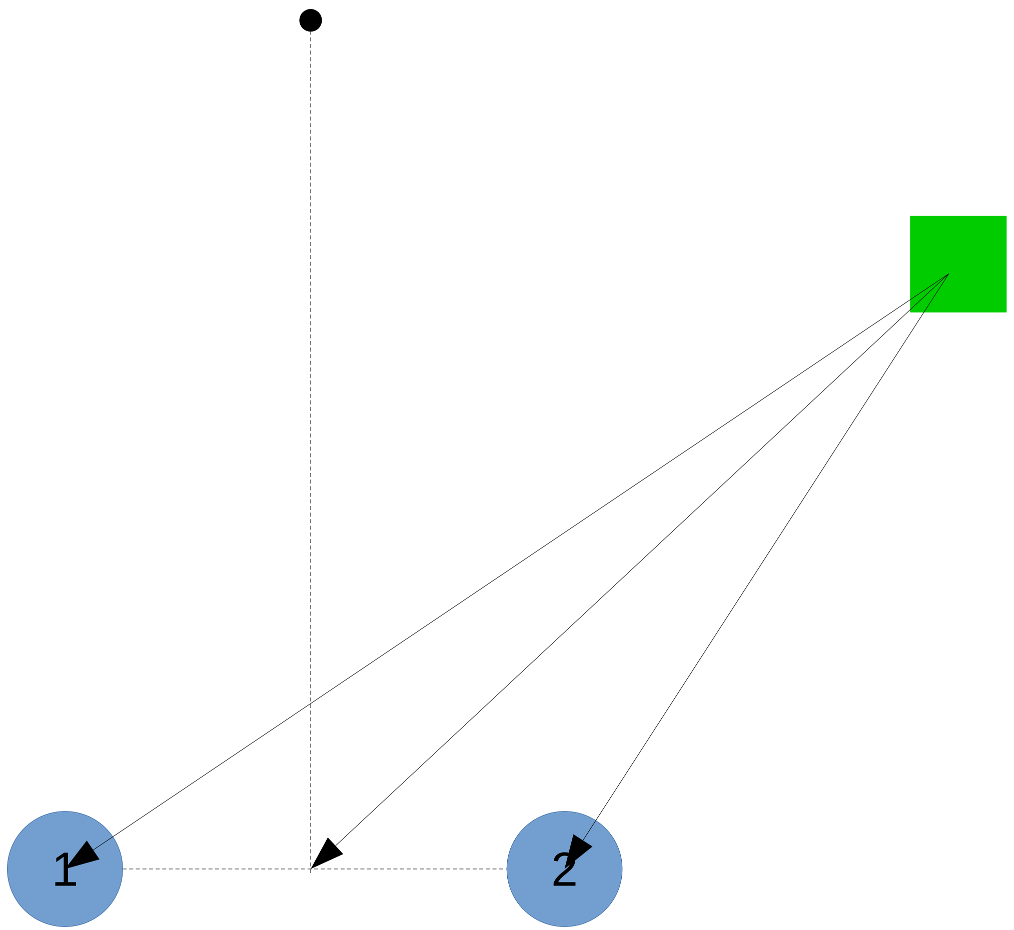
Modelo “Campo Cercano” de Sonido



Modelo “Campo Cercano” de Sonido

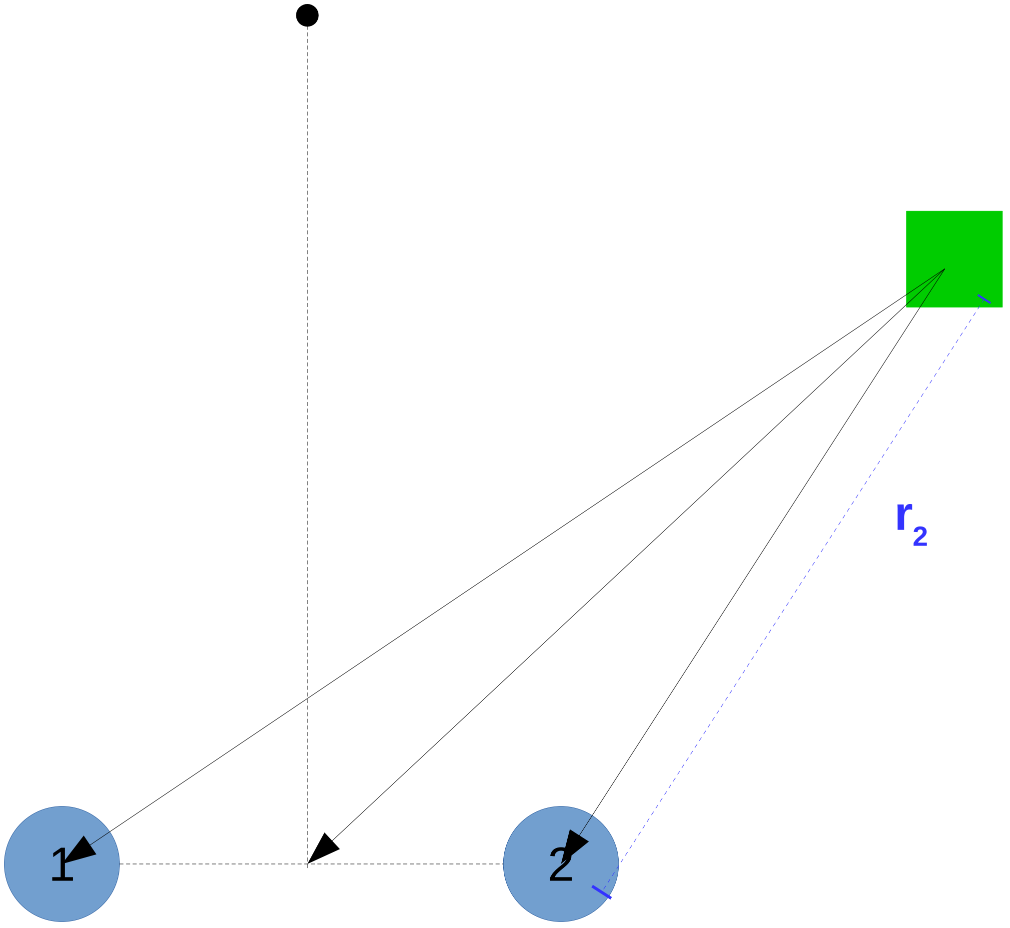


Modelo “Campo Cercano” de Sonido

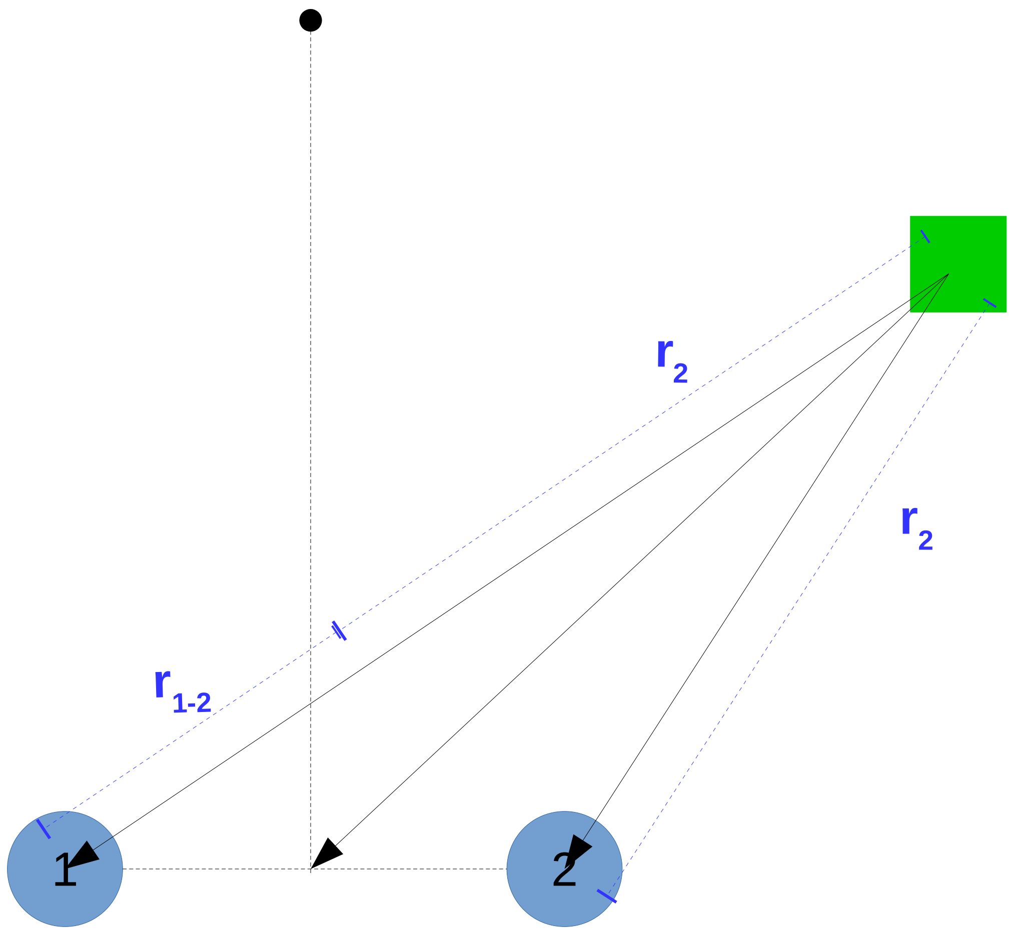


Llega primero a 2 que a 1...

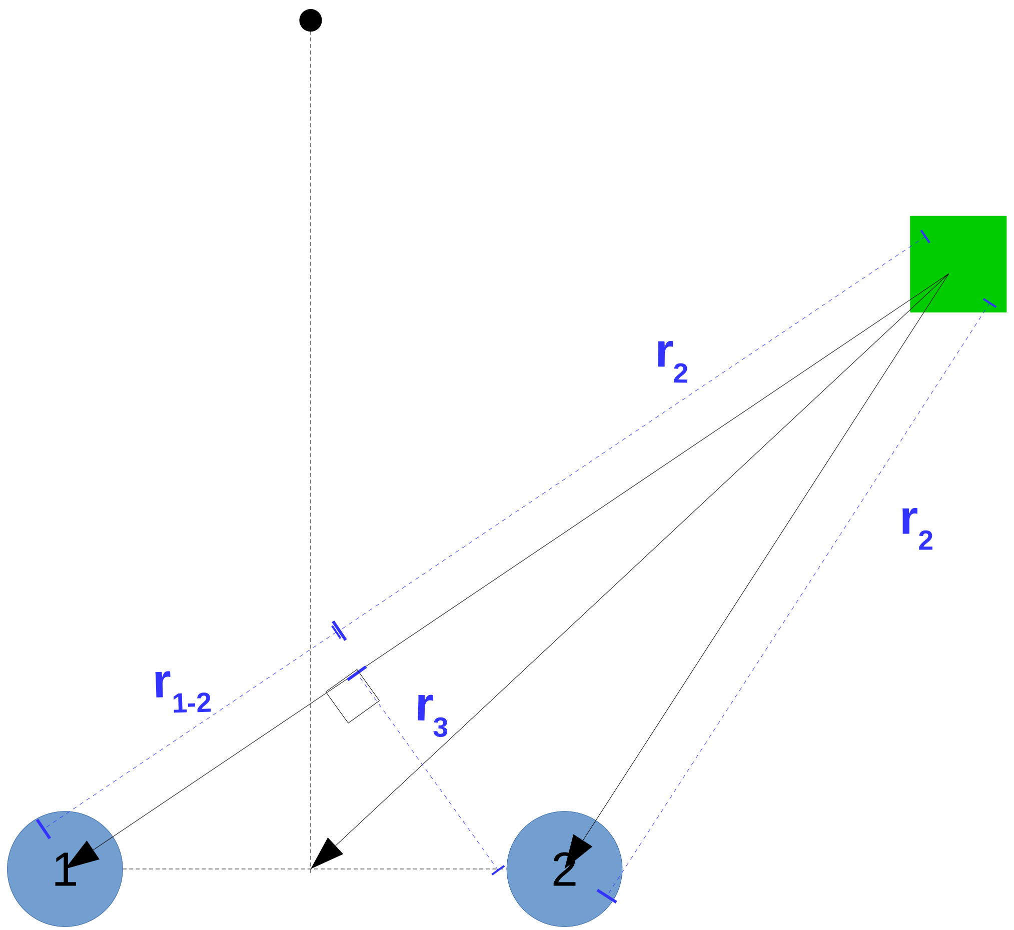
Modelo “Campo Cercano” de Sonido



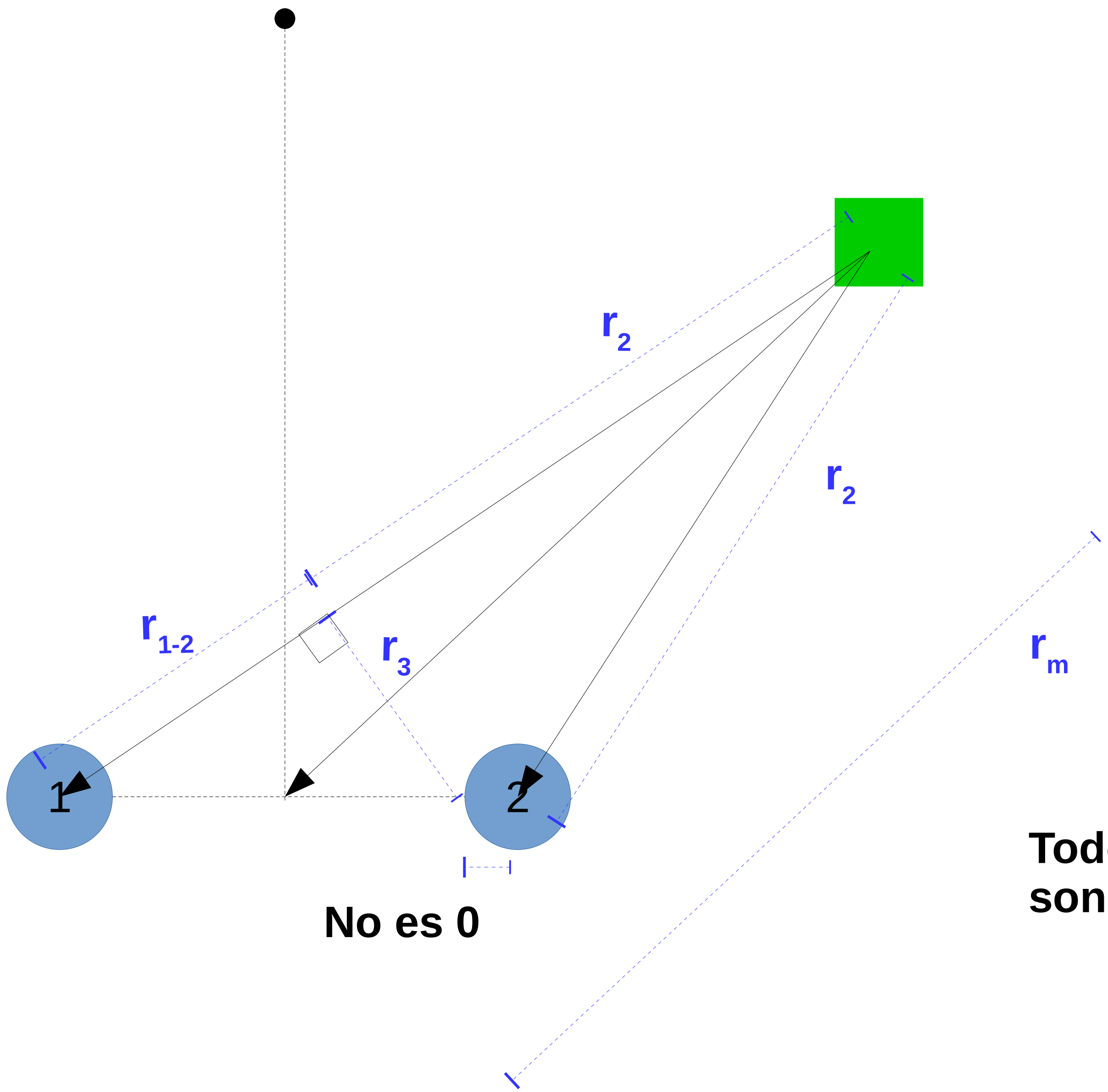
Modelo “Campo Cercano” de Sonido



Modelo “Campo Cercano” de Sonido

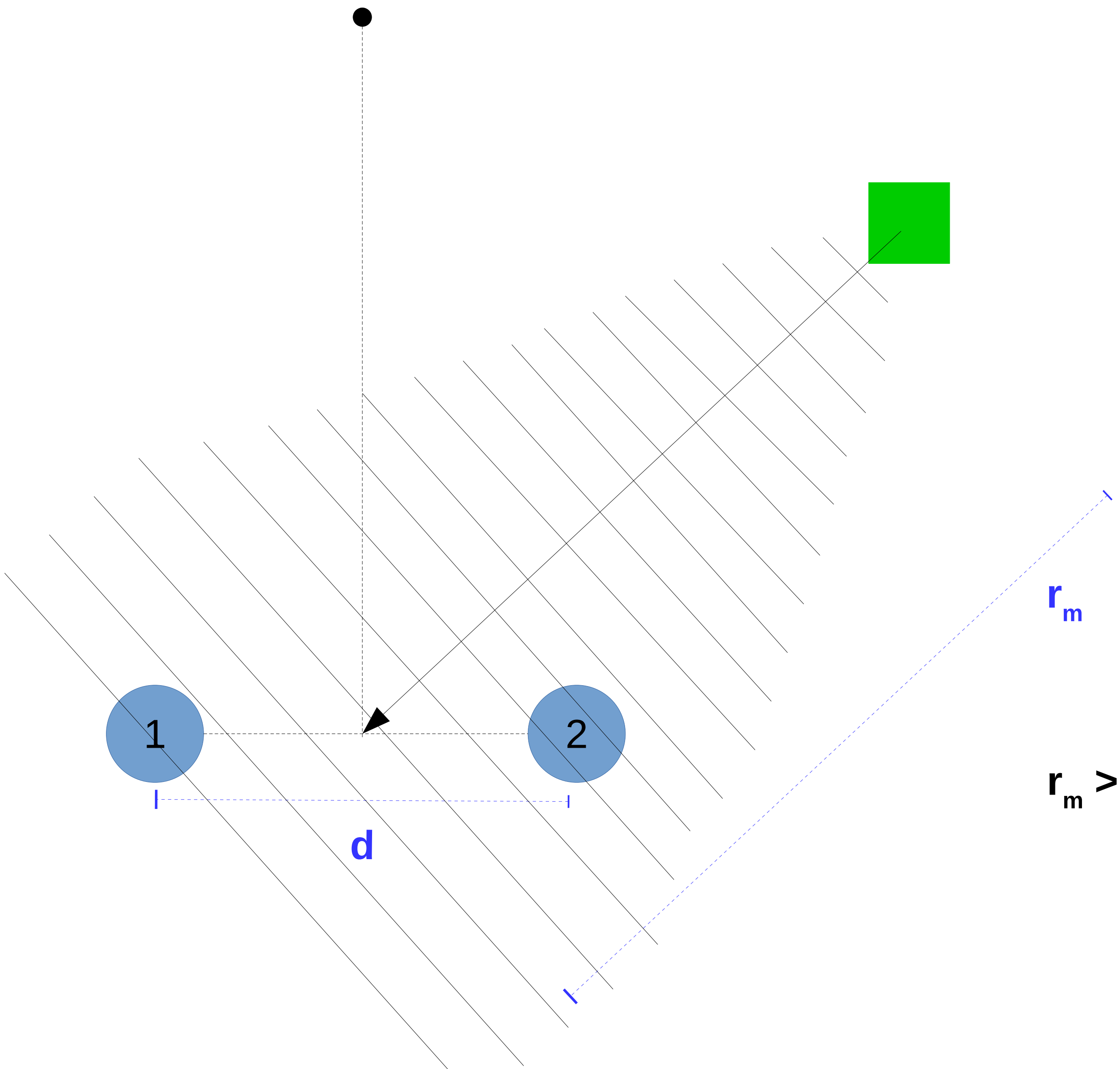


Modelo “Campo Cercano” de Sonido



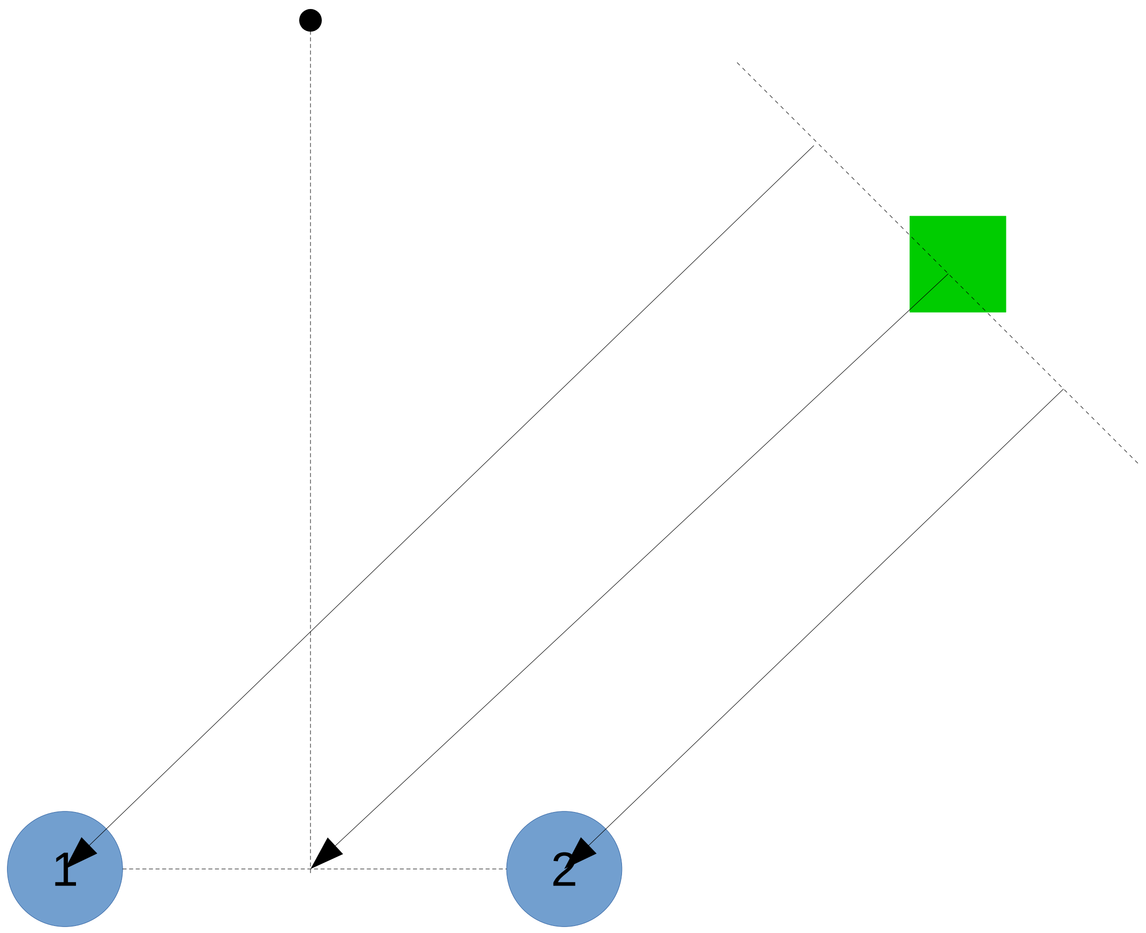
Todos los *r's* son desconocidos

Modelo “Campo *Lejano*” de Sonido

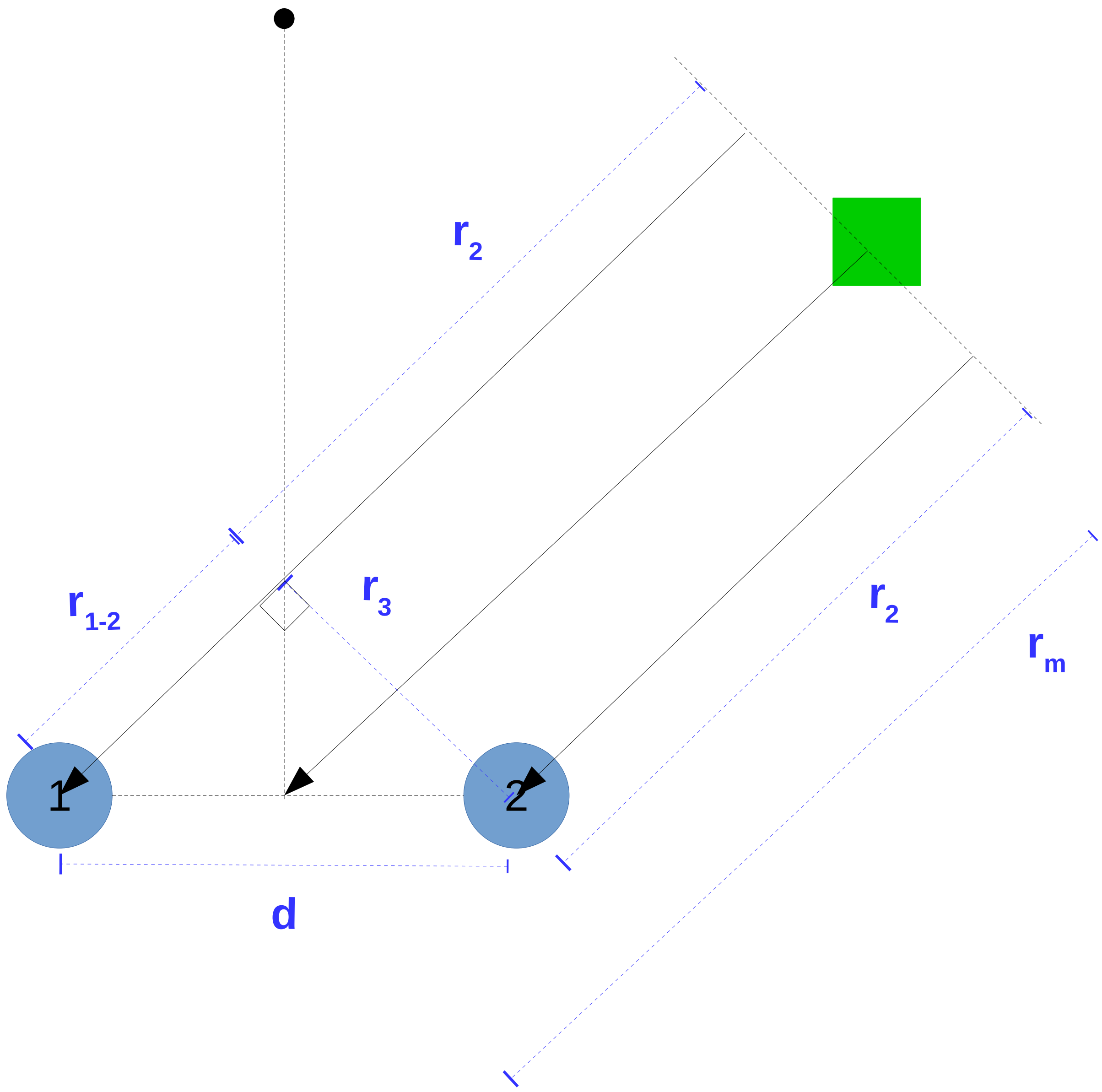


$r_m >>> d$

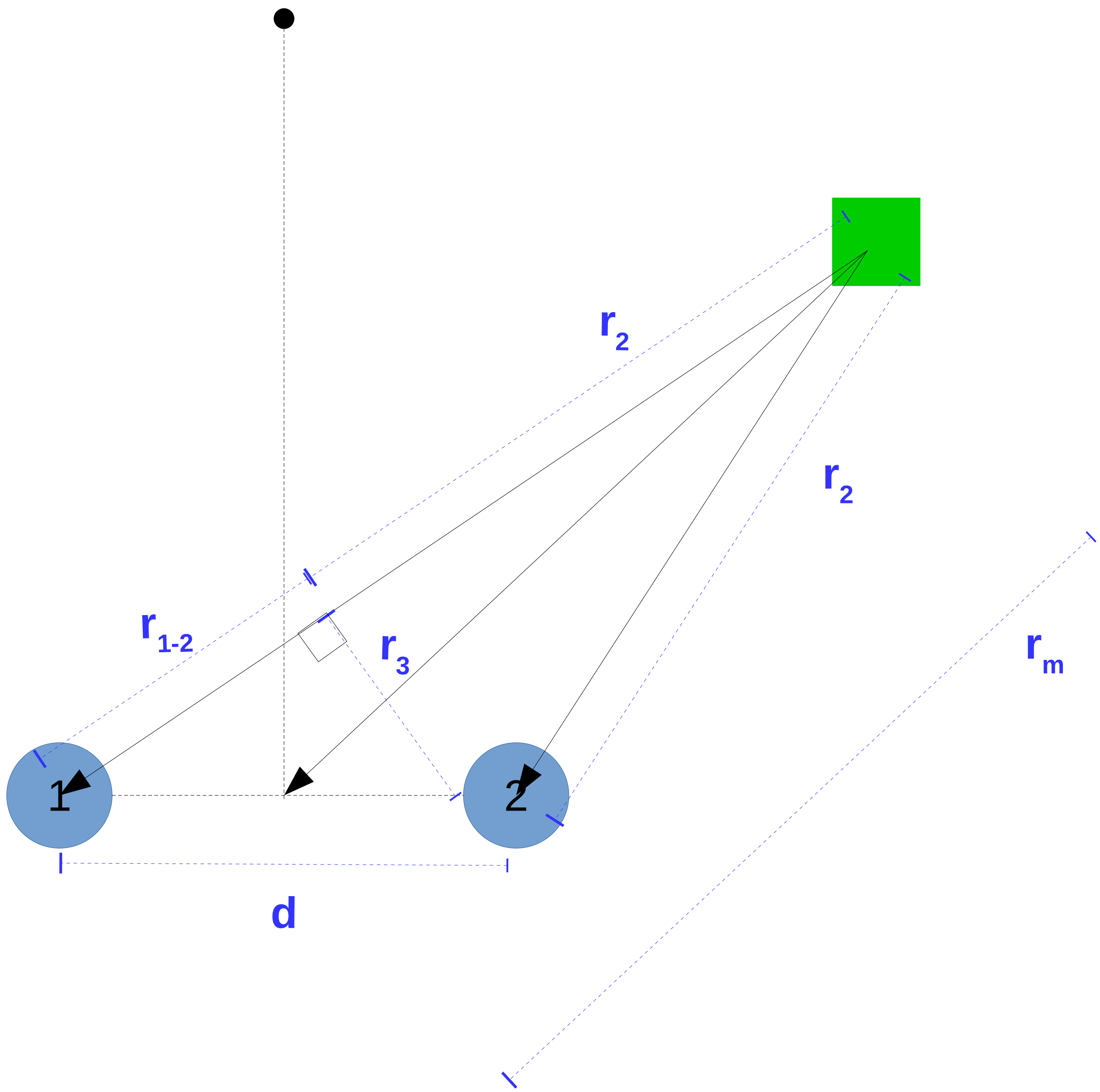
Modelo “Campo *Lejano*” de Sonido



Modelo "Campo *Lejano*" de Sonido

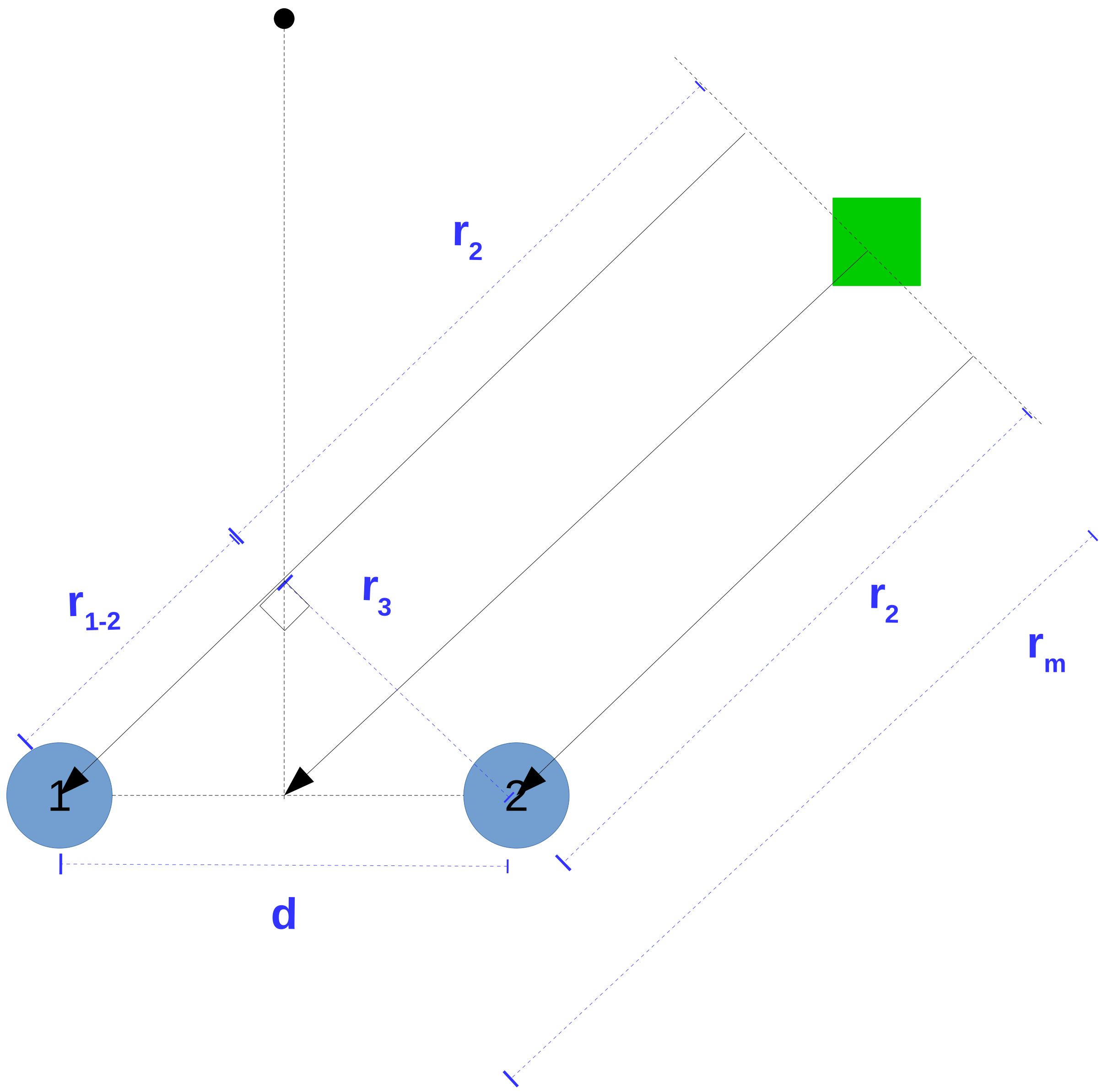


Modelo “Campo Cercano” de Sonido



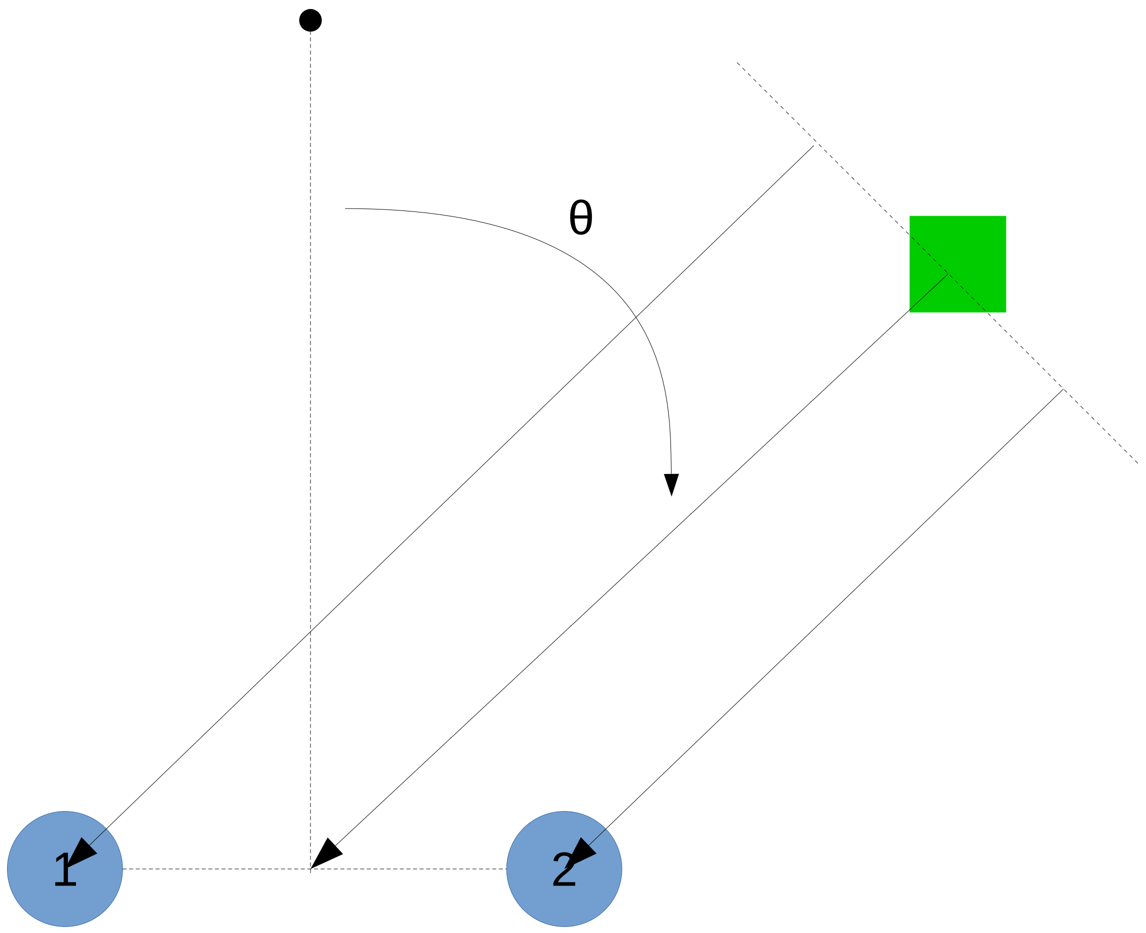
Modelo “Campo *Lejano*” de Sonido

¿Para que?



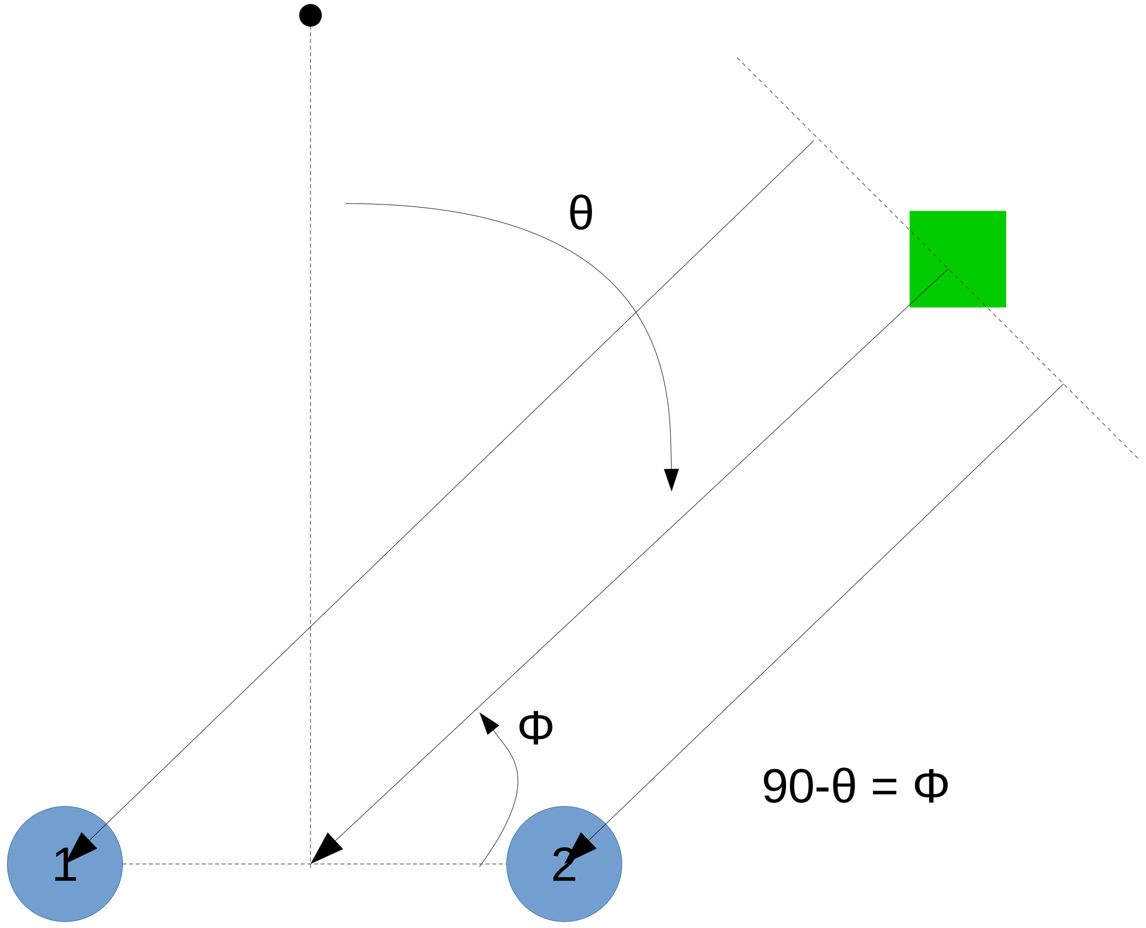
Modelo “Campo *Lejano*” de Sonido

¿Para que?



Modelo “Campo *Lejano*” de Sonido

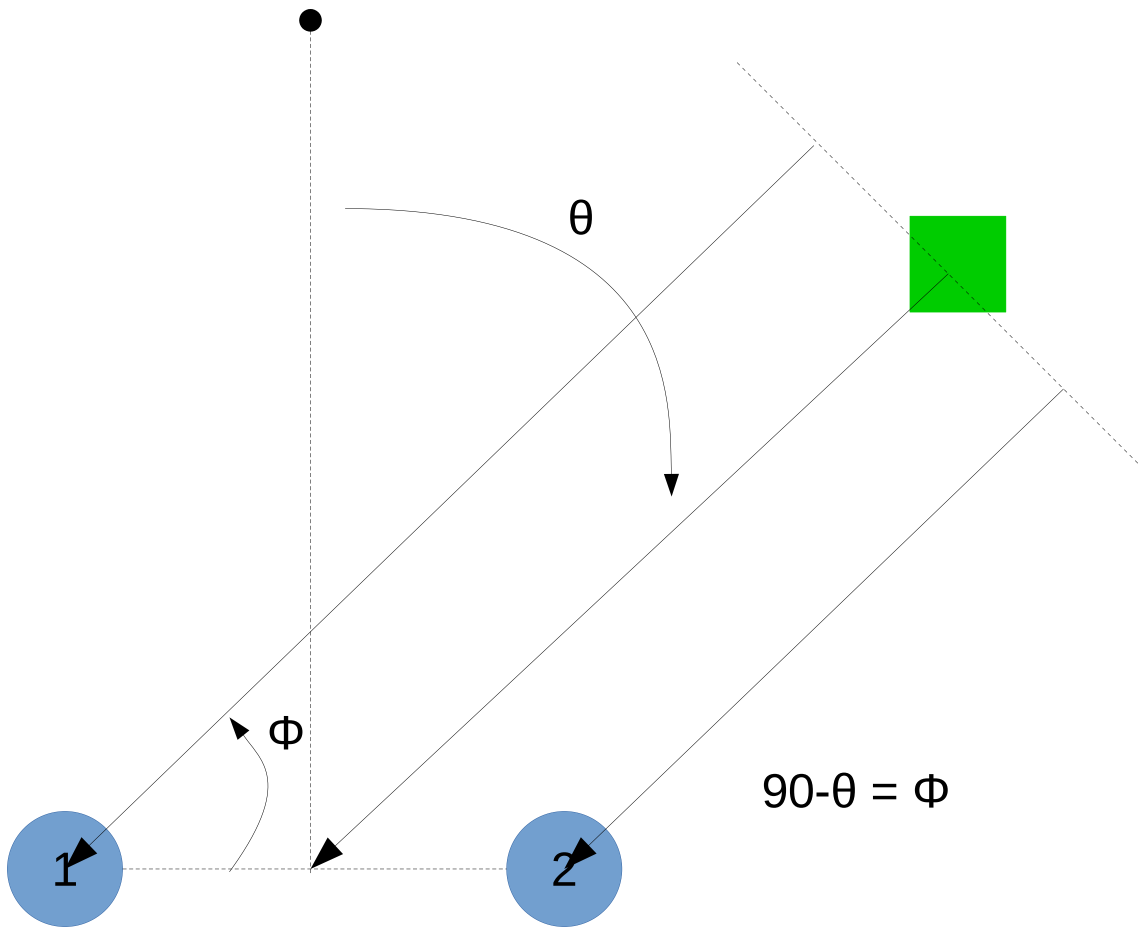
¿Para que?



90-θ = Φ

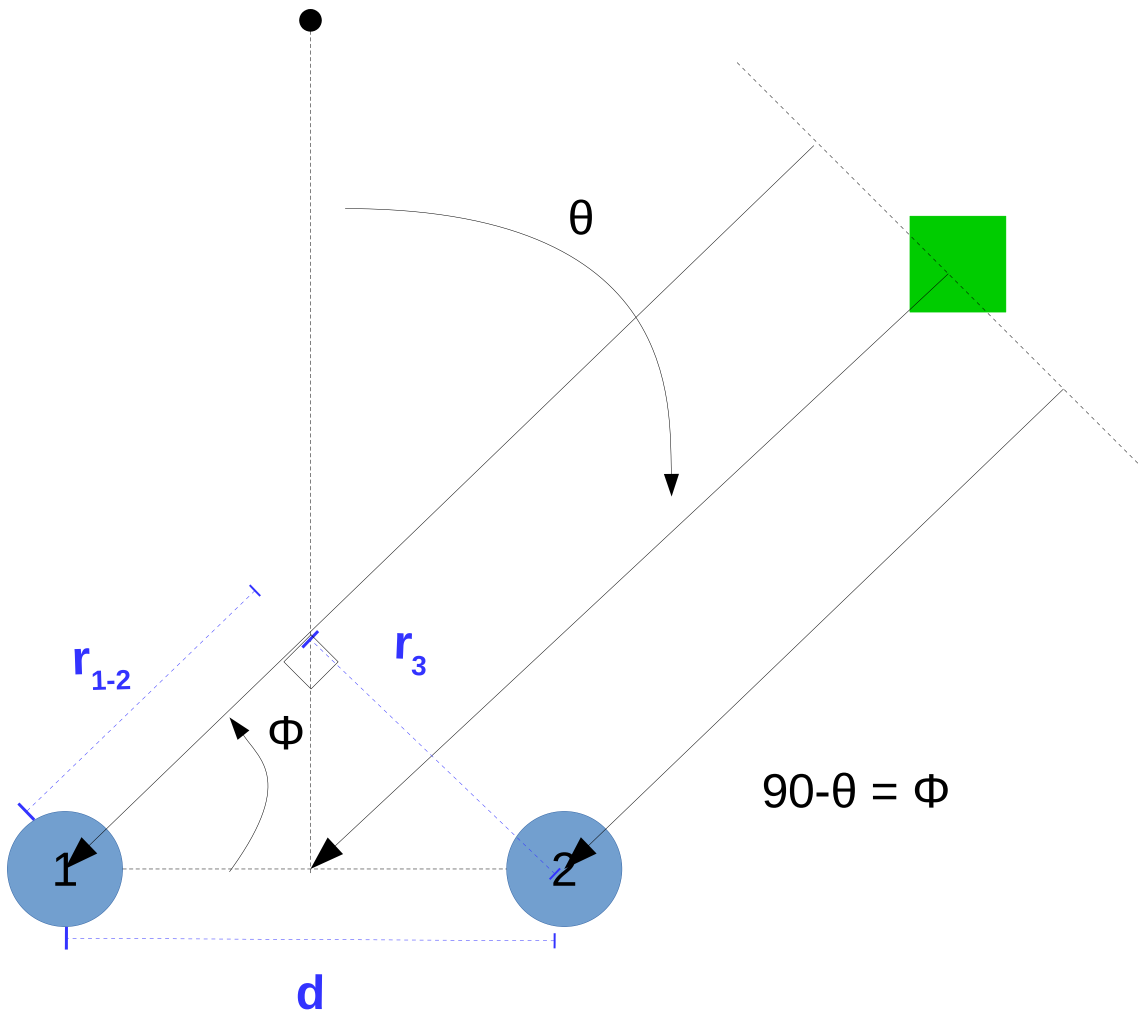
Modelo “Campo *Lejano*” de Sonido

¿Para que?



Modelo “Campo *Lejano*” de Sonido

¿Para que?



90-θ = Φ

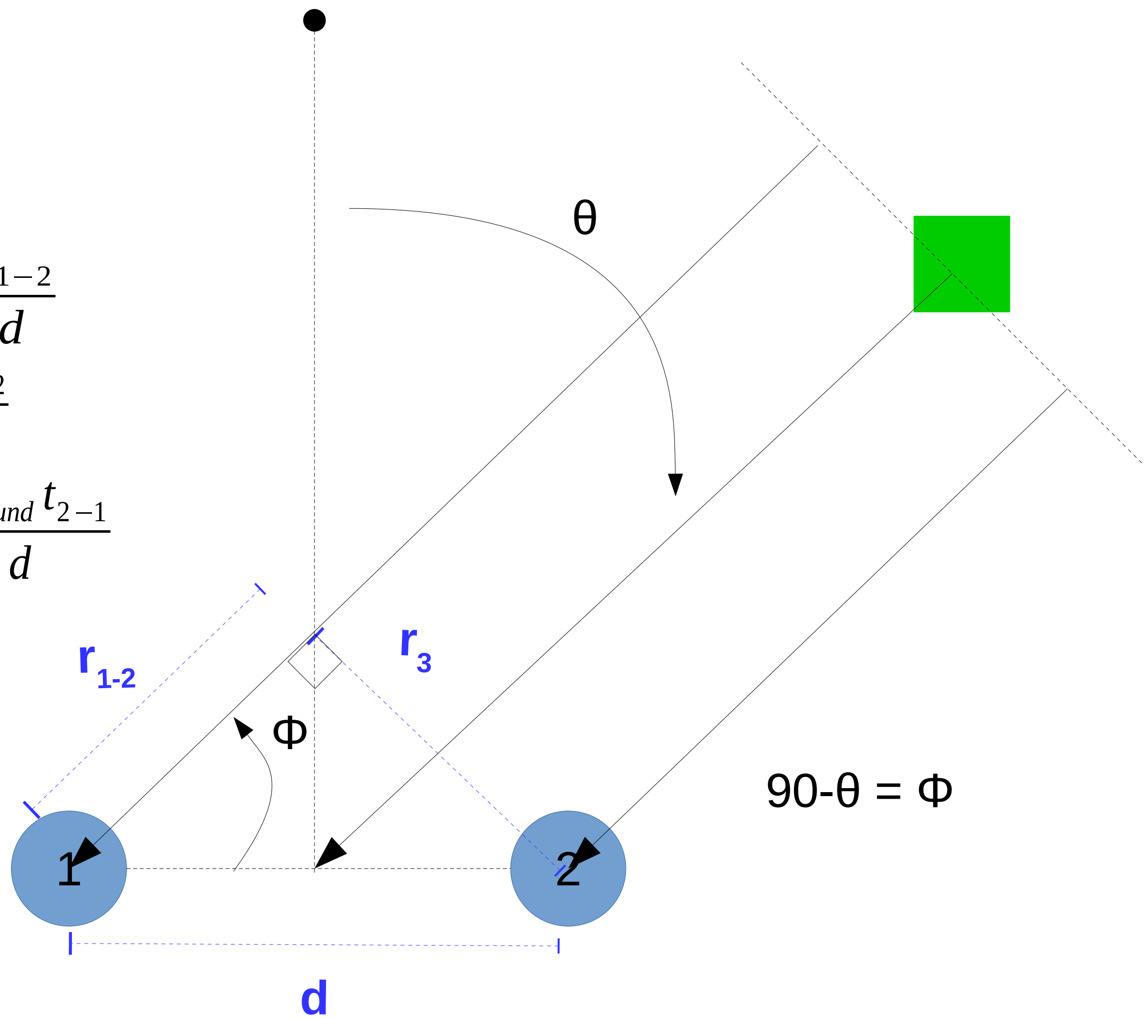
Modelo “Campo *Lejano*” de Sonido

¿Para que?

$$\cos(\Phi) = \frac{r_{1-2}}{d}$$

$$\sin(\theta) = \frac{r_{1-2}}{d}$$

$$= \frac{V_{sound}\, t_{2-1}}{d}$$

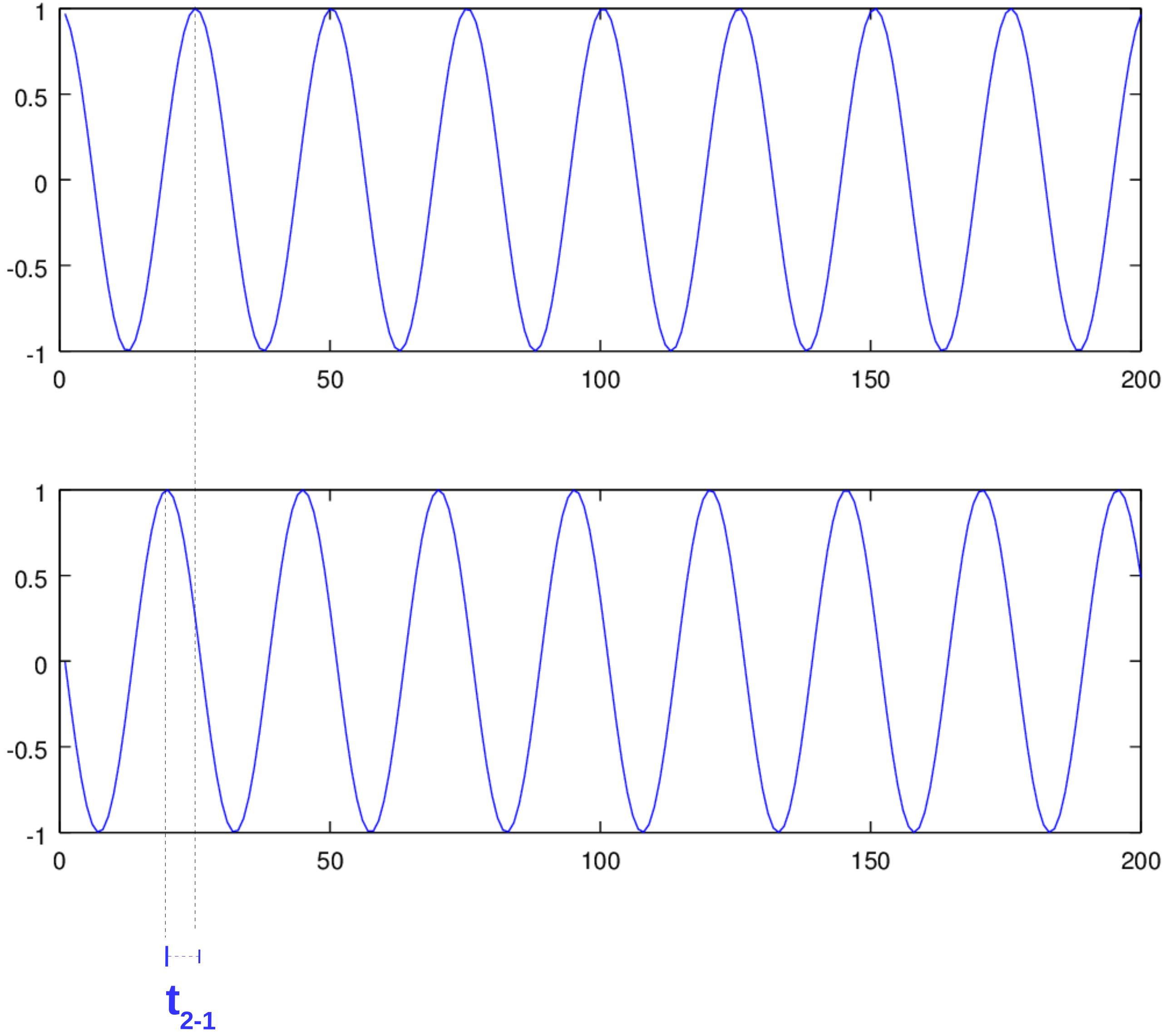


t_{2-1} = Tiempo (en segundos) que dura el sonido llegar a 1, dado que ya haya llegado a 2
:= **desfase** de la señal capturada en 2 vs la capturada en 1

Desfase

Señal 1

Señal 2



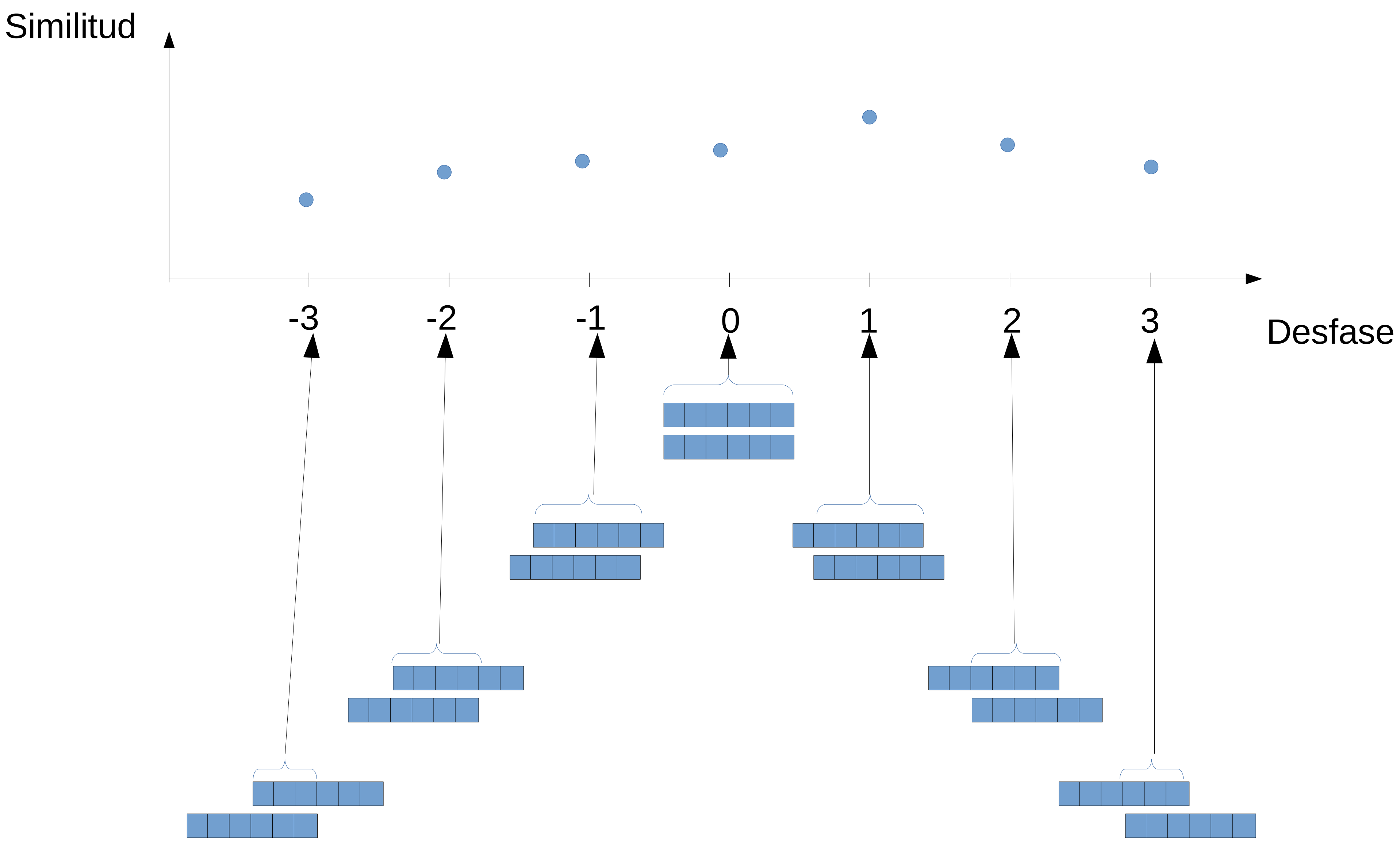
Desfase

- Si podemos encontrar el desfase entre las señales, podemos utilizarlo para calcular la Dirección de Arribo.
- ¿Pero cómo?

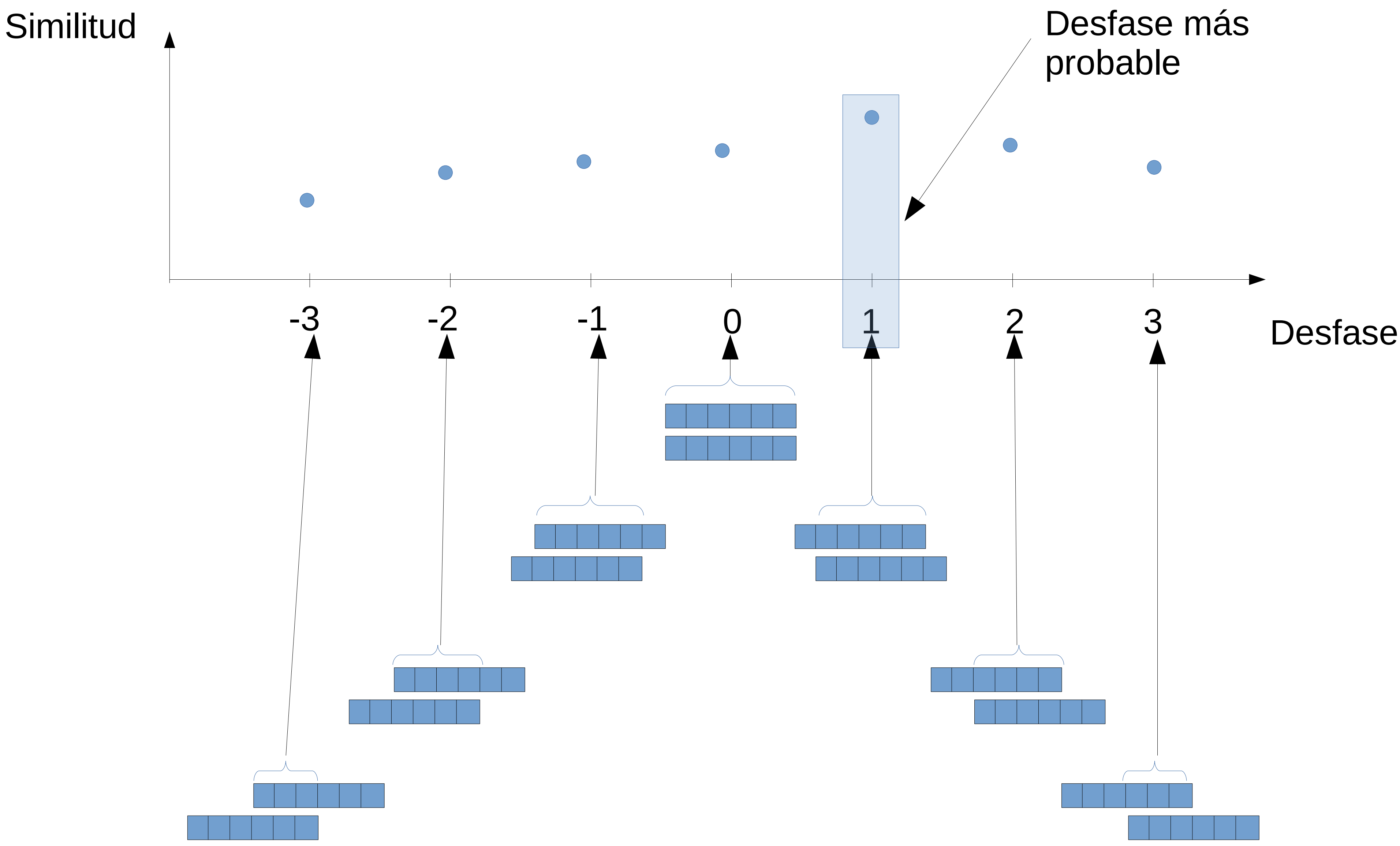
Correlación Cruzada

- Básicamente:
  - Desfasar una señal.
  - Compararlas
- Por cada desfase posible, registrar su “similitud”.
- Y así obtener, el vector de correlación cruzada.
  - También conocido como correlograma.
    - En inglés: “*cross-correlogram*”

Vector de Correlación Cruzada

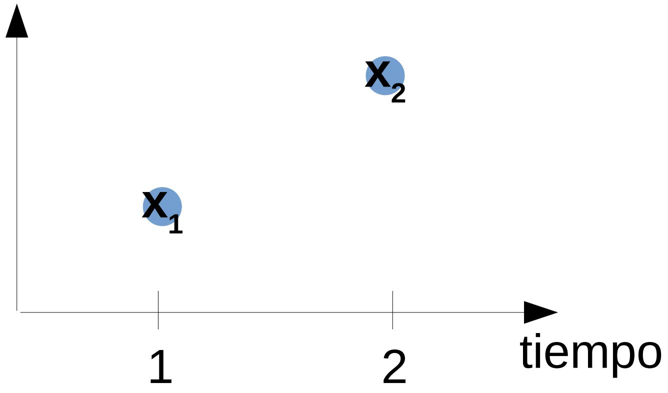


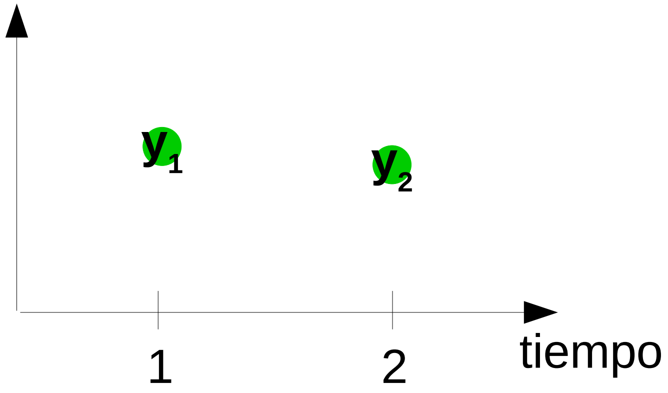
Vector de Correlación Cruzada



Similitud

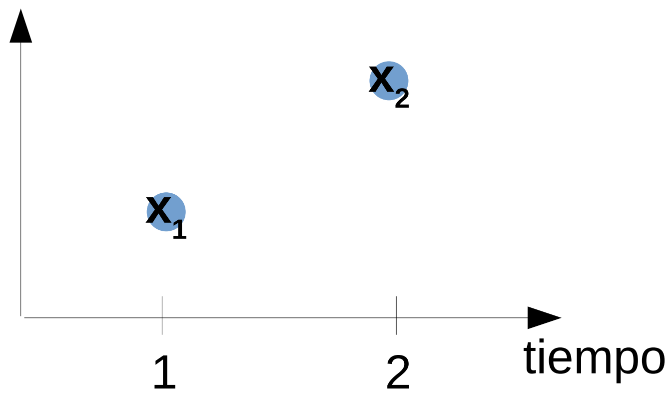
Digamos que tenemos dos señales, con dos muestras (*frames*) cada una.

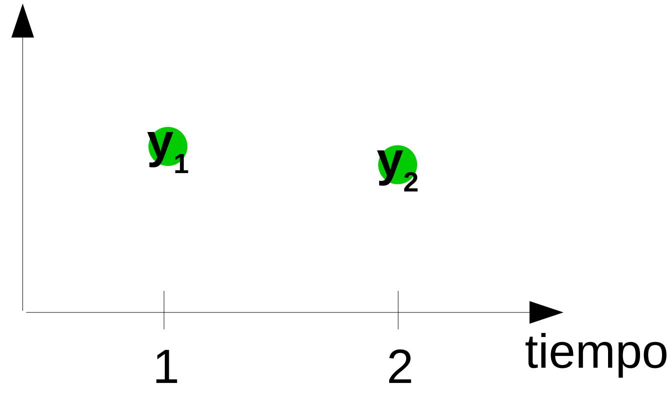




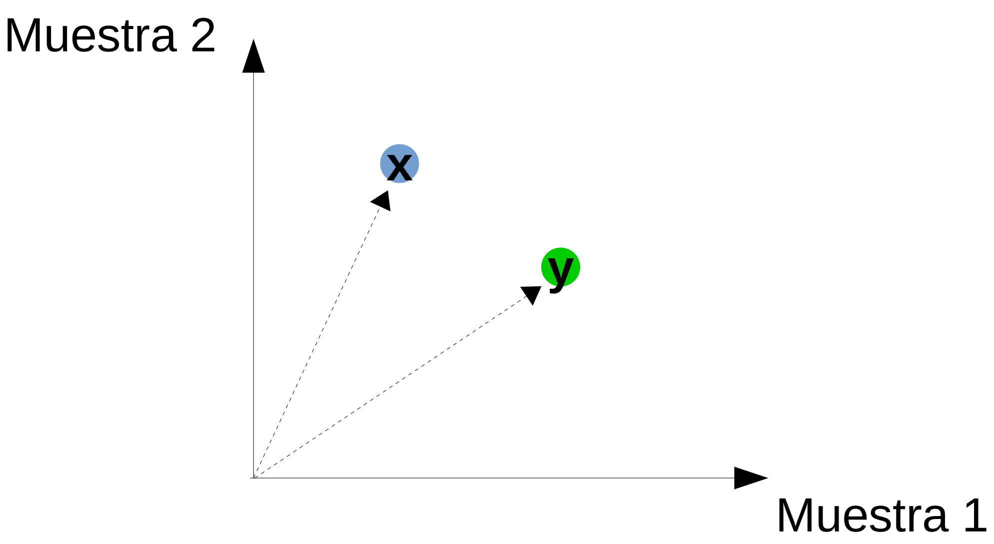
Similitud

Digamos que tenemos dos señales, con dos muestras (*frames*) cada una.

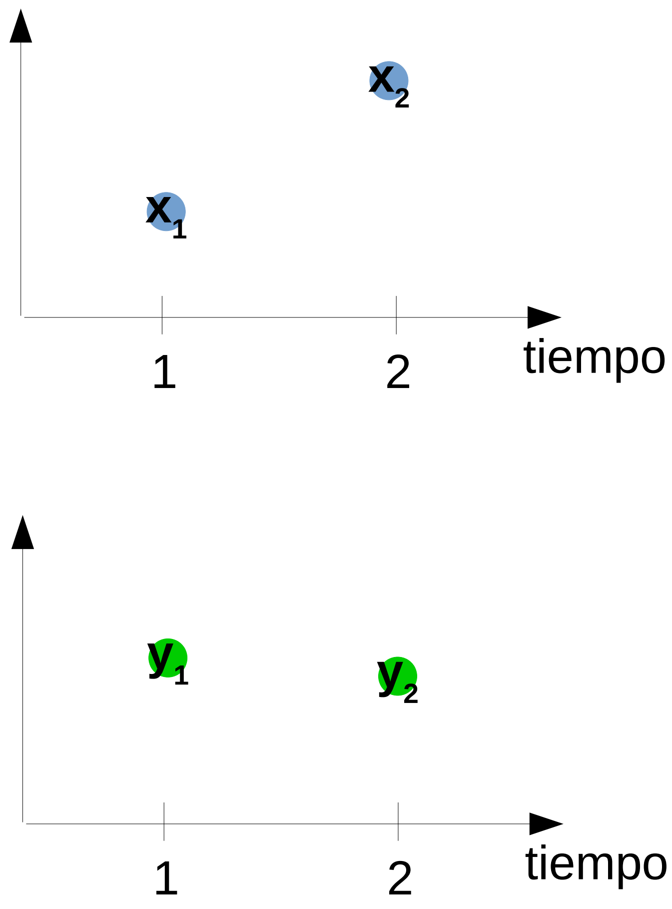




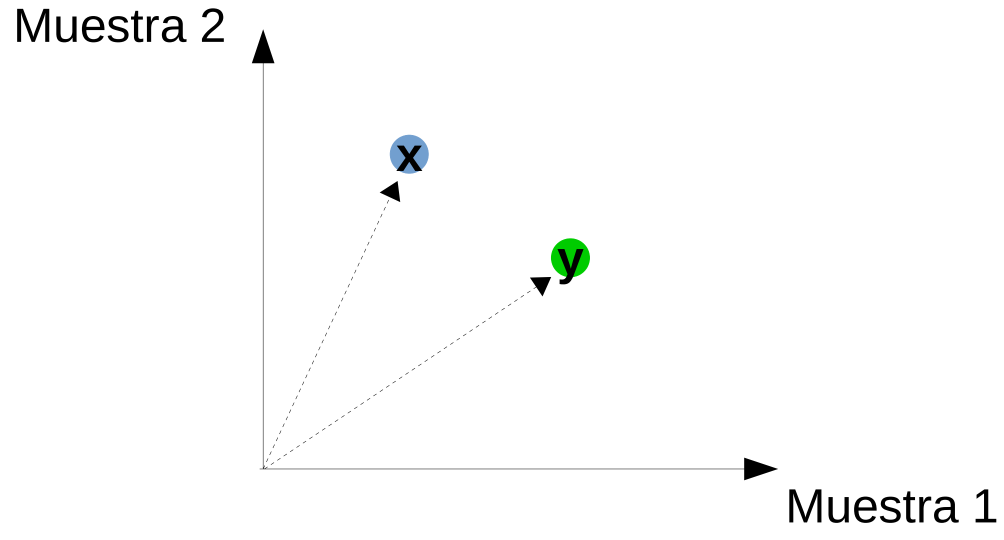
Cada señal se puede presentar como un vector de dos dimensiones (una dimensión por cada muestra).



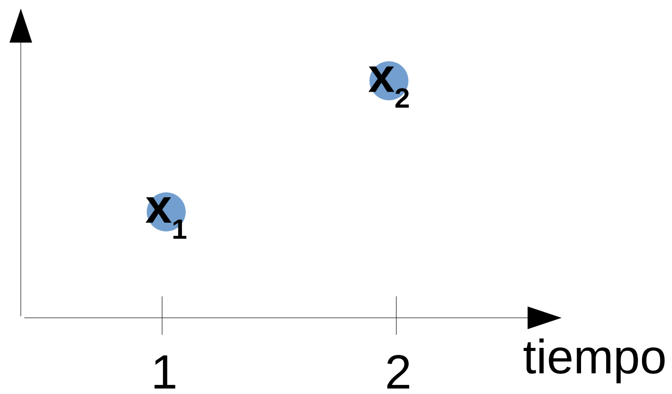
Similitud

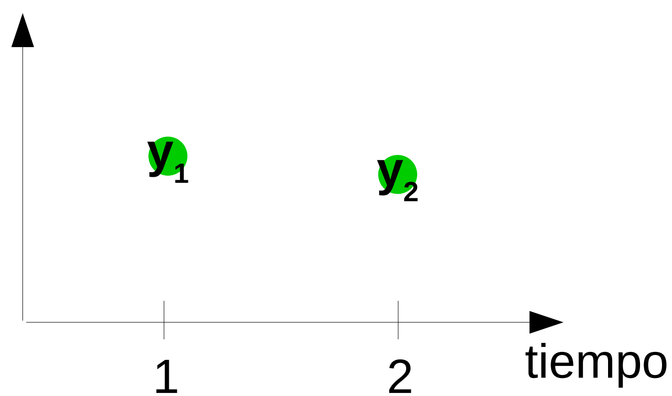


Se puede mostrar la similitud entre estos dos vectores de diversas formas:



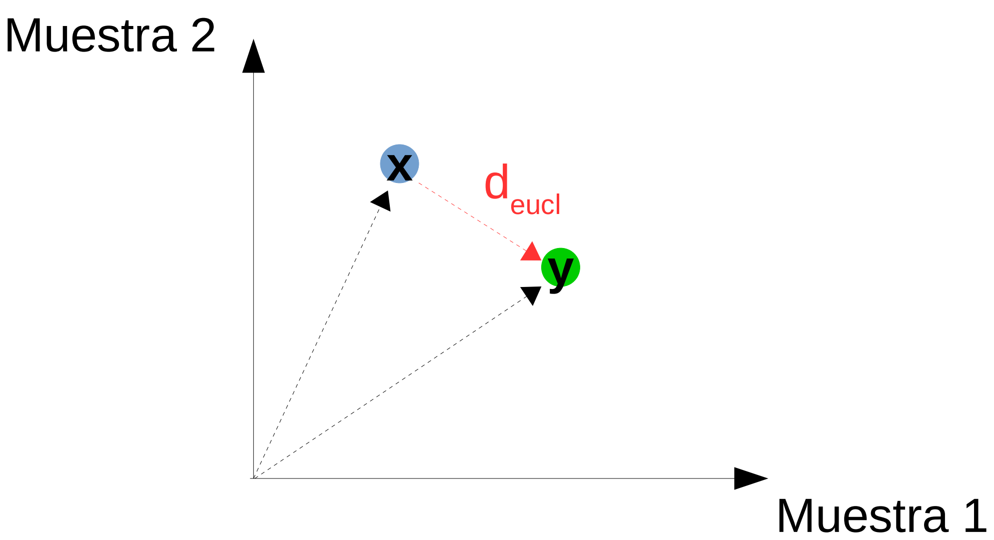
Similitud





Distancia Euclidiana:

$$d_{eucl} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$



$d_{eucl} = 0$

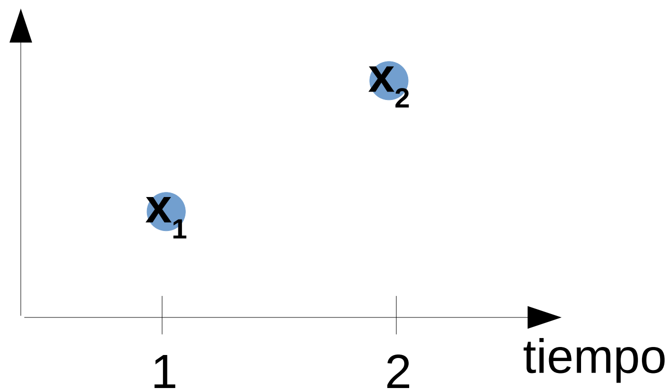
:= señales idénticas

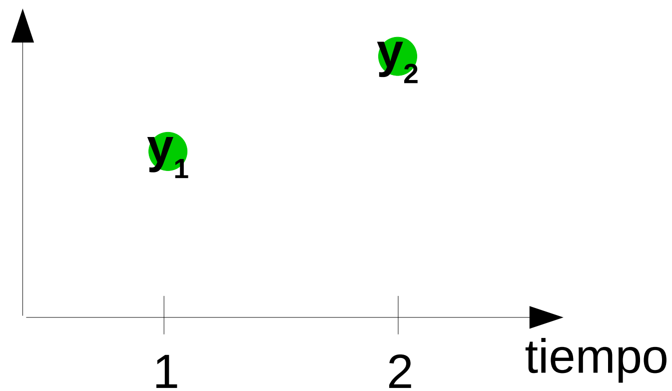
Distancia Euclidiana

- Tiene problemas en mostrar similitud cuando hay señales que son similares, pero con diferente magnitud.
- Por ejemplo:

Similitud

Son similares
(diferente magnitud)

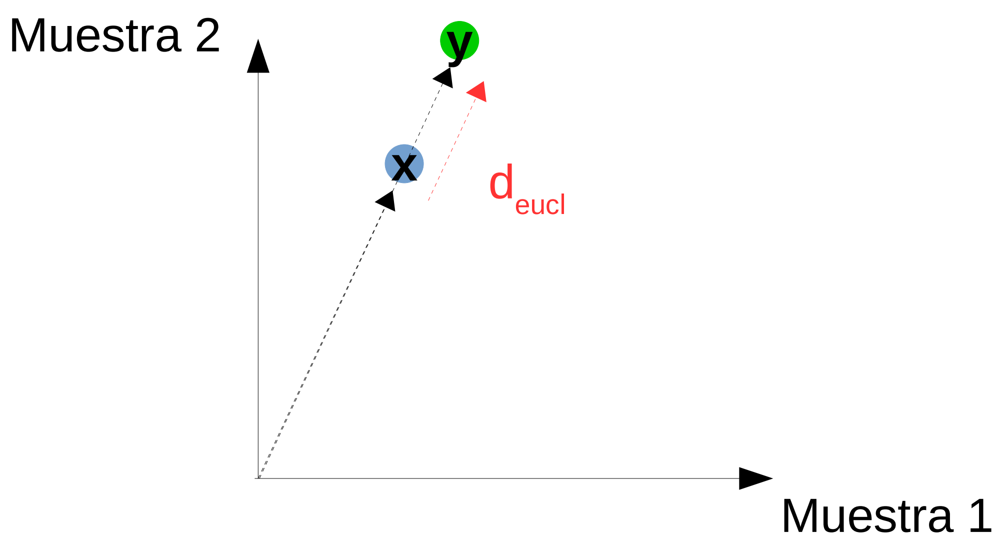




d_{eucl} != 0
:= señales no-idénticas

Distancia Euclidiana:

$$d_{eucl} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$



Distancia Euclidiana

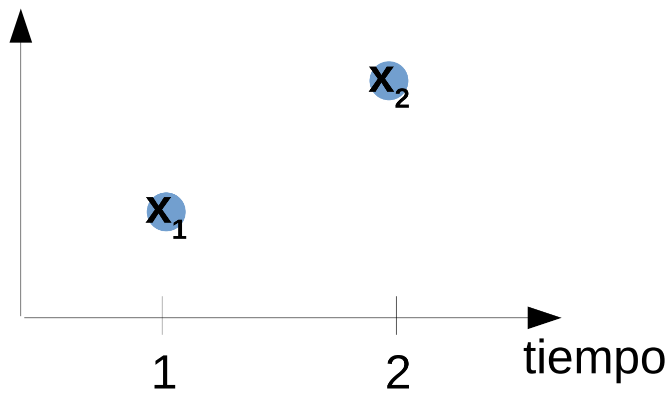
- Es buena en presentar si las señales son idénticas.
- No es tan buena para presentar cuando son “similares”.
  - Que tienen la **misma forma**, aun cuando tiene diferente magnitud.
  - Esto es importante, ya que es muy posible que un micrófono esté más lejos que el otro de la fuente, haciendo que reciba la señal con diferente energía.

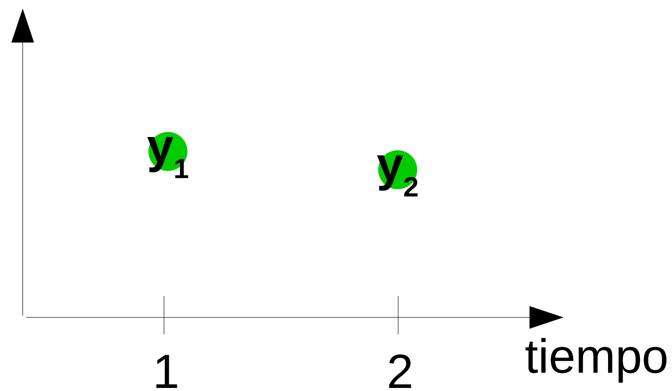
¿Entonces?

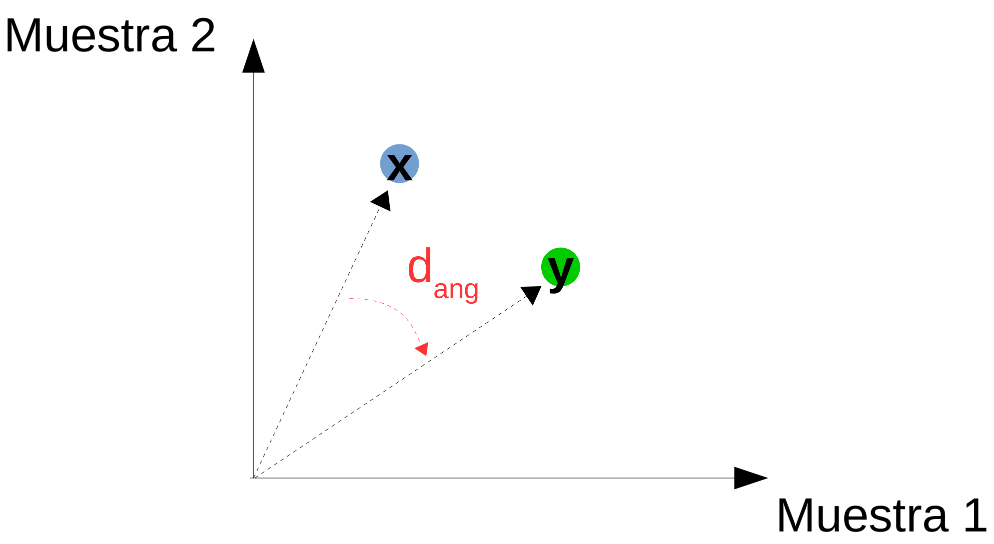
Similitud

$d_{ang} = 0$

:= señales similares

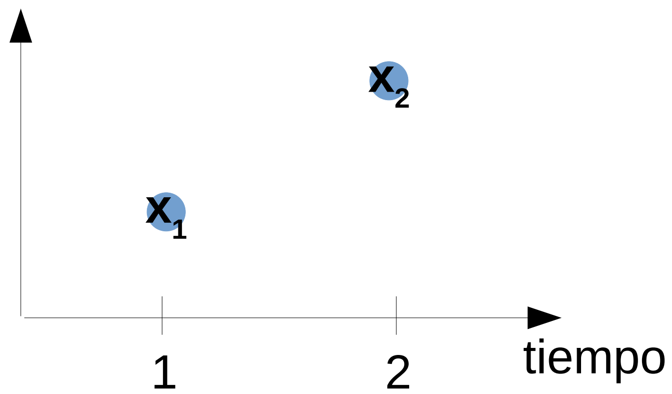


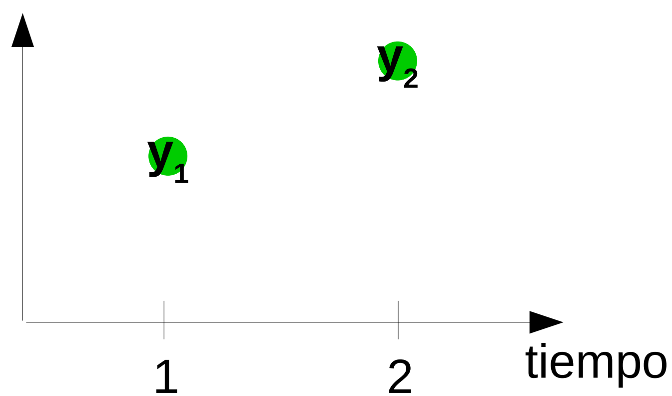




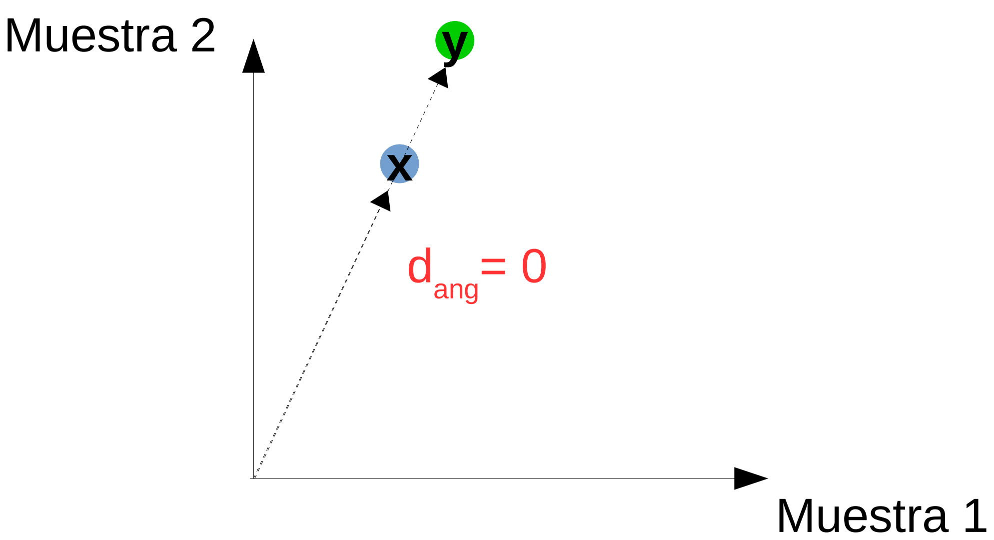
Similitud

Son similares
(diferente magnitud)





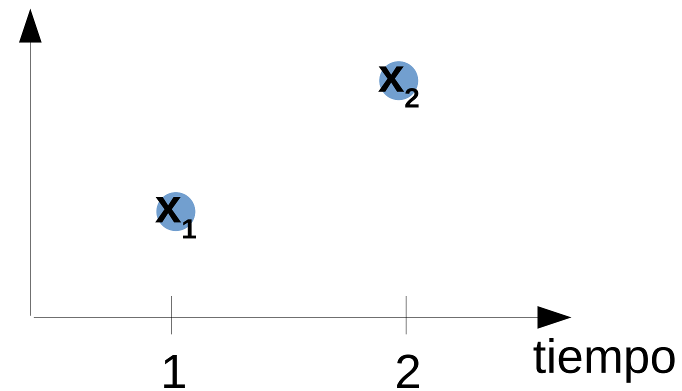
$d_{ang} = 0$
:= señales similares



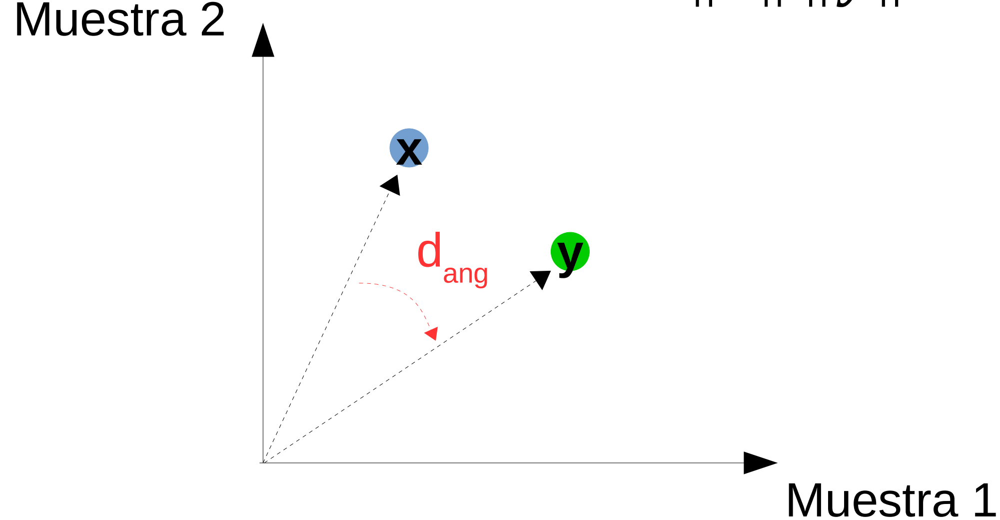
Angulo entre Dos Vectores

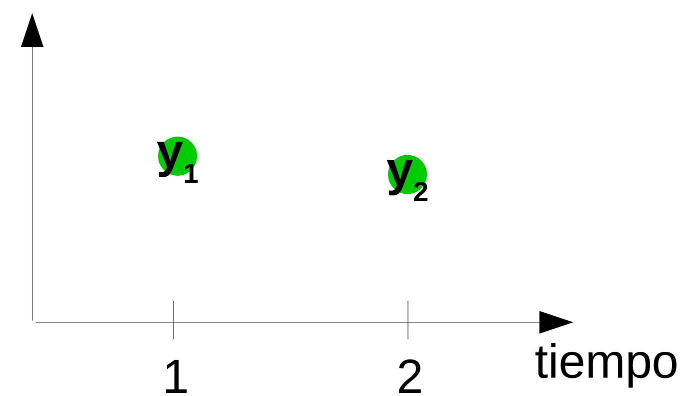
$$d_{ang} = \arccos\left(\frac{x_1 y_1 + x_2 y_2}{\sqrt{x_1^2 + x_2^2}\sqrt{y_1^2 + y_2^2}}\right)$$

$$\cos(d_{ang}) = \frac{x_1 y_1 + x_2 y_2}{\sqrt{x_1^2 + x_2^2}\sqrt{y_1^2 + y_2^2}} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$



$$\cos(d_{ang}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\|\boldsymbol{x}\| \, \|\boldsymbol{y}\|} = r$$





$r = 1$ $\quad (d_{ang} = 0)$

:= señales “similares”

:= señales **correlacionadas**

Coeficiente Pearson

- Más bien dicho: el *Coeficiente de Producto-Momento de Pearson*
- Mide la correlación entre señales, sin importar la magnitud.
  - Una forma de verlo: porque está ligado al ángulo entre las señales.
  - Otra forma de verlo: porque está normalizando el producto entre las señales por sus magnitudes.

$$\cos\left(d_{ang}\right) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\|\boldsymbol{x}\| \, \|\boldsymbol{y}\|} = r$$

Correlación Cruzada

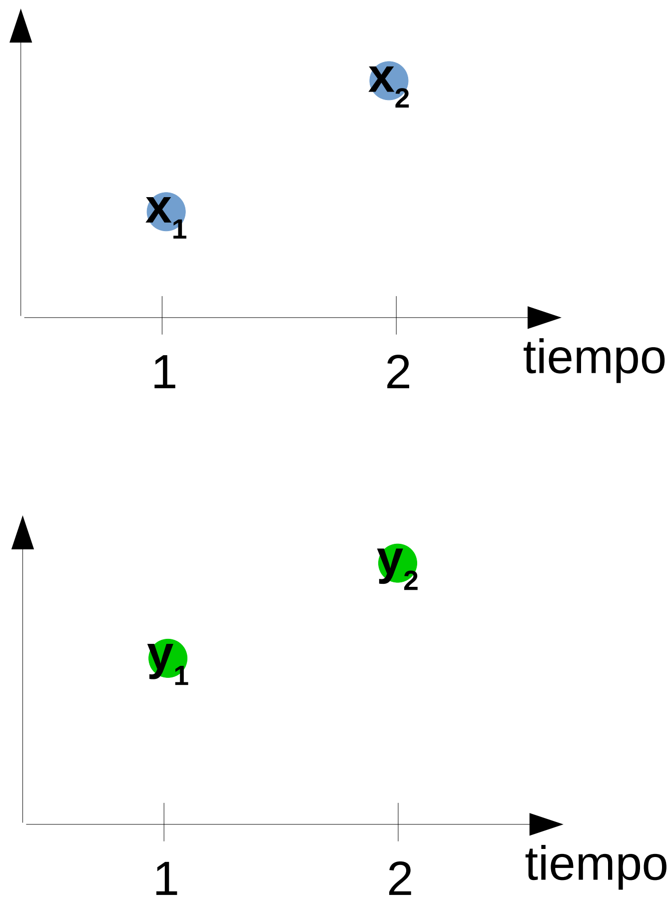
- De hecho, es referida como “Producto Punto Deslizante”.
  - Porque se “desliza” (o desfasa) una señal, en cada desfase, se calcula el producto punto de ambas señales.
- Al utilizar el Coeficiente Pearson, la única diferencia es que estamos haciendo una Correlación Cruzada *Normalizada*.

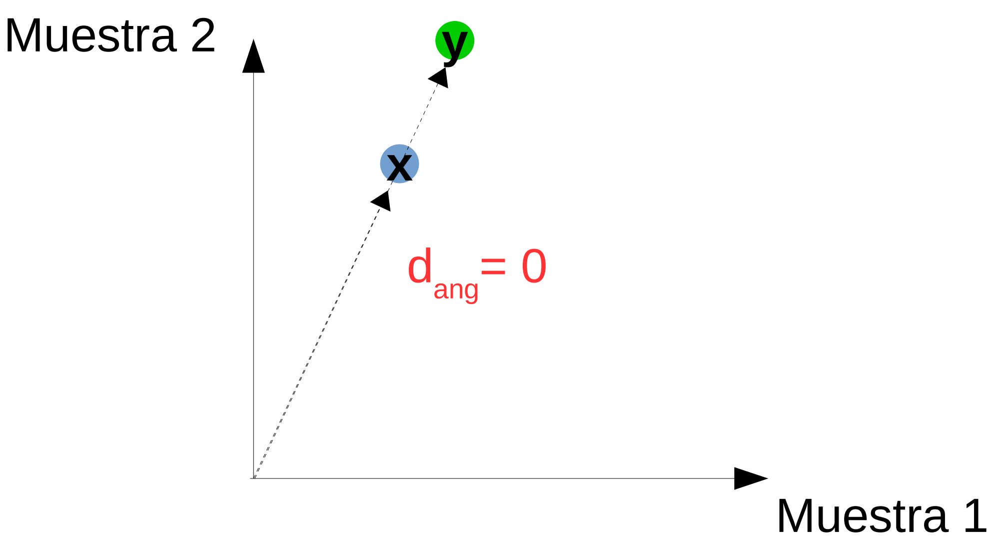
Propiedad de Pearson

- El $\cos(d_{ang})$ puede tener valores entre -1 a 1.
- $r > 0$:= correlación positiva
  - Cuando la primera señal incrementa su energía, también la segunda.
- $r < 0$:= correlación negativa
  - Cuando la primera señal incrementa su energía, la segunda *decrece* su energía.
- $r = 0$:= no hay correlación
  - Cuando la primera señal incrementa su energía, la segunda *a veces* decrece su energía, *a veces* incrementa.

Similitud

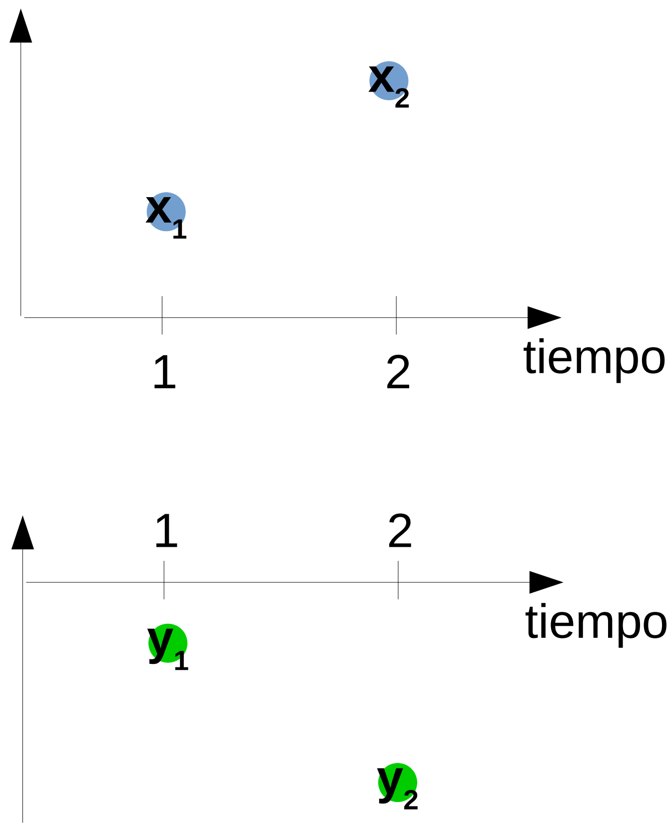
$d_{ang} = 0$
$r > 0$

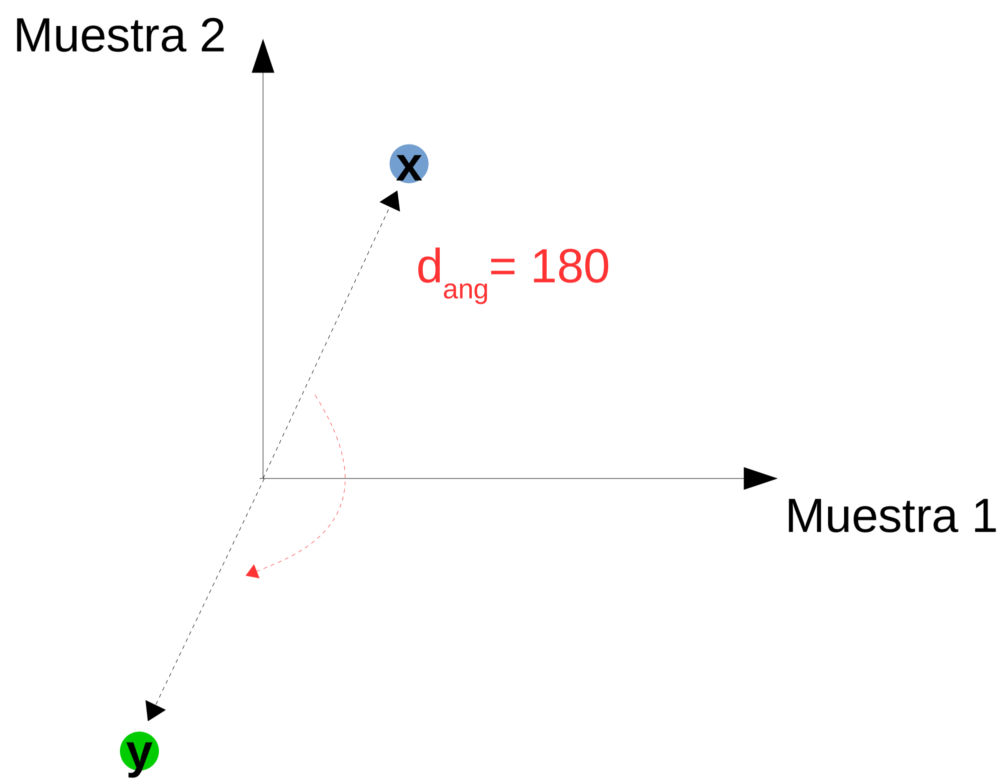




Similitud

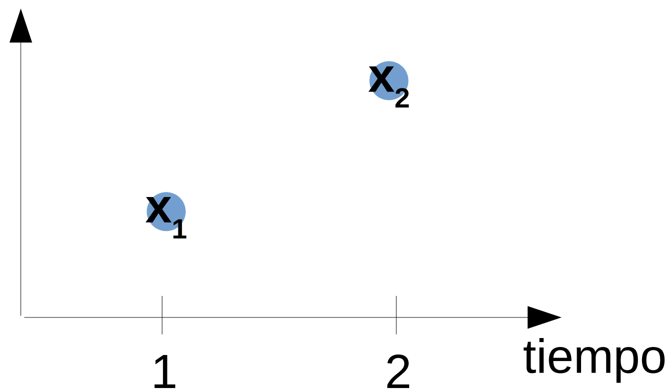
$d_{ang} = 180$
$r < 0$

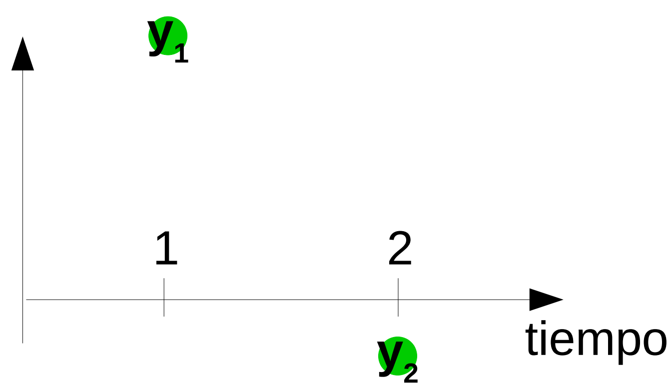


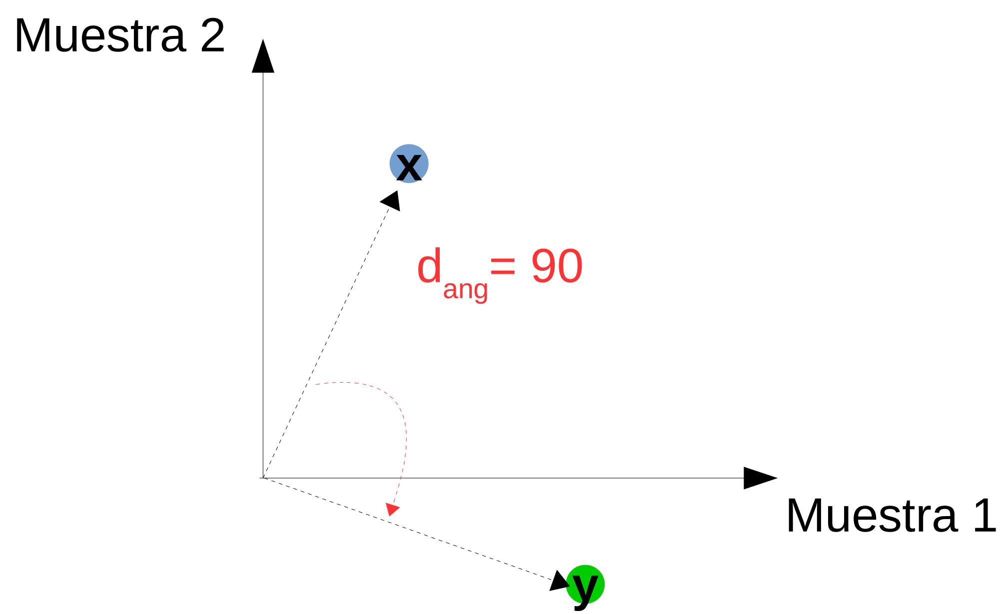


Similitud

$d_{ang} = 90$
$r = 0$







Pero...

- Con r = 0, la señal también esta decreciendo.
  - ¿No es esto una correlación negativa?
- En este caso, es un poco difícil visualizarlo porque sólo son dos muestras. Tendríamos que ver si en otros momentos incrementa también.
- Pero, si las señales estuvieran **centradas**, esto se puede visualizar perfectamente:

Señales Centradas

- Es importante que, al momento de calcular el coeficiente Pearson entres dos señales, ambas estén **centradas**.
  - Calcular el promedio, y luego restárselo a todas sus muestras.
  - Dícese:

$$r = \frac{\sum_{i=1}^{n} (x_i - \overline{\boldsymbol{x}})(y_i - \overline{\boldsymbol{y}})}{\sqrt{\sum_{i=1}^{n} (x_i - \overline{\boldsymbol{x}})^2} \sqrt{\sum_{i=1}^{n} (y_i - \overline{\boldsymbol{y}})^2}}$$

Señales Centradas

- En términos de estadística: al centrar las señales, el coeficiente Pearson se convierte en la **covariancia** de la señales, dividido entre el producto de sus desvaciones estándar.
  - Esto va a ser importante recordarlo cuando lleguemos a Múltiples DOA.

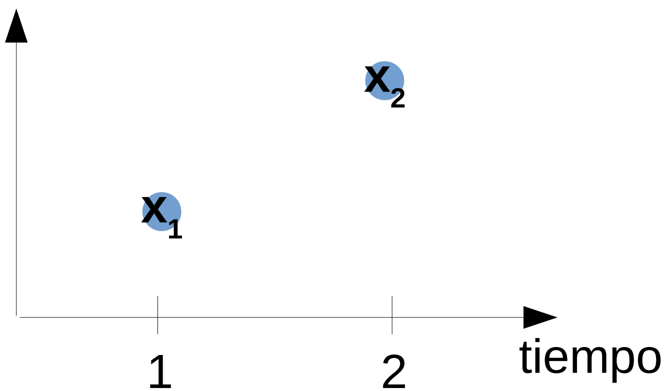
$$r=\frac{\sum_{i=1}^{n}(x_i-\boldsymbol{\bar{x}})(y_i-\boldsymbol{\bar{y}})}{\sqrt{\sum_{i=1}^{n}(x_i-\boldsymbol{\bar{x}})^2}\sqrt{\sum_{i=1}^{n}(y_i-\boldsymbol{\bar{y}})^2}}$$

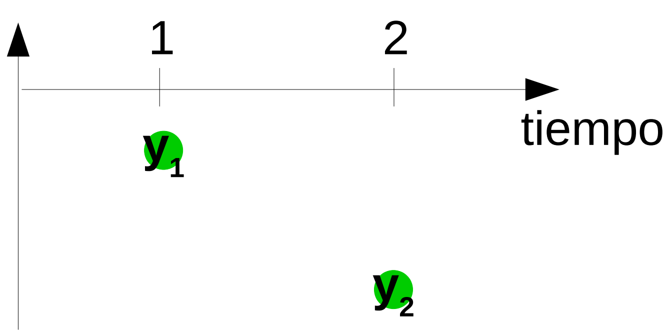
Entonces, si asumimos que las señales están centradas...

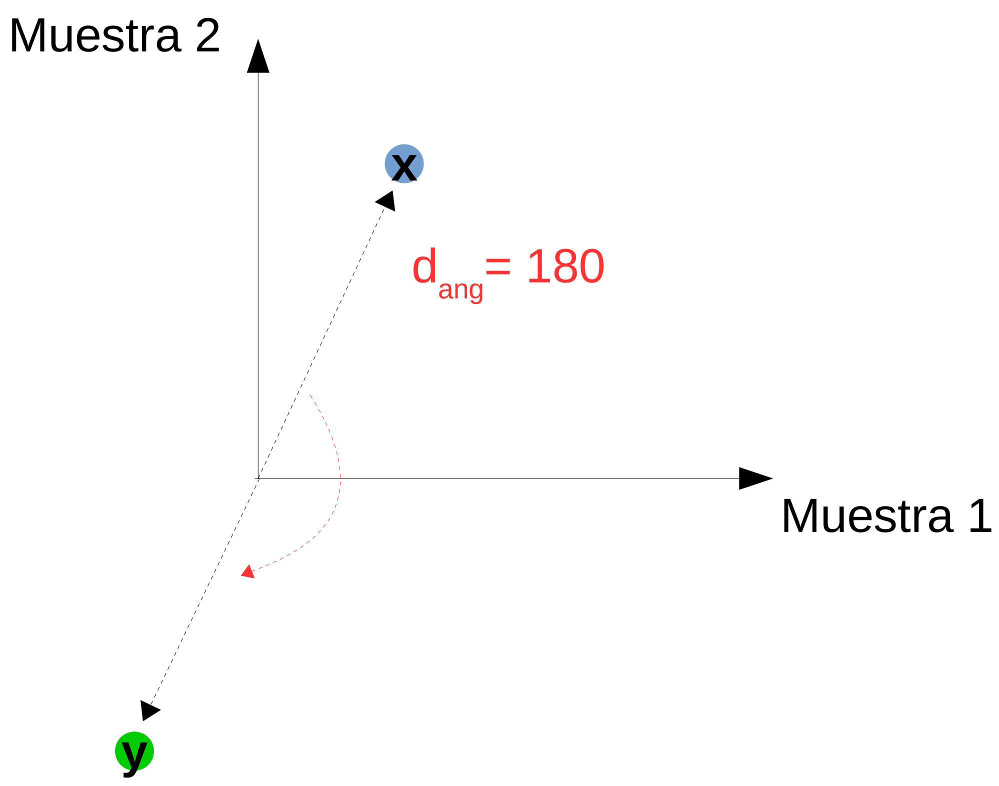
Similitud

Asumiendo que las señales están centradas:
“y” siempre se mantiene en la parte negativa, y decreciendo.
Clara correlación negativa ante “x” que se mantiene en la parte positiva, y creciendo.

$d_{ang} = 180$
$r < 0$



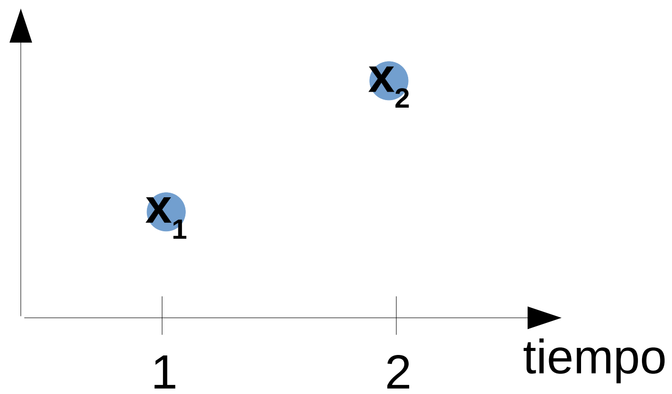


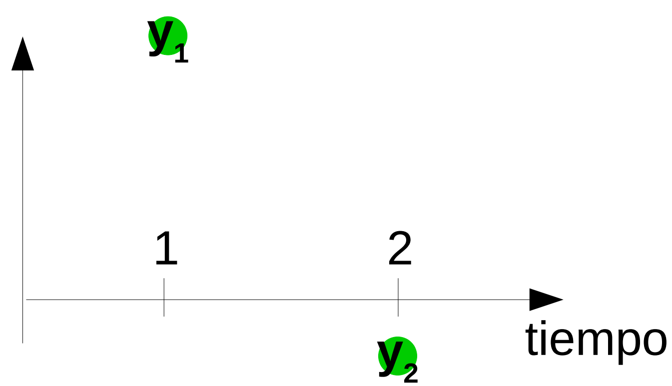


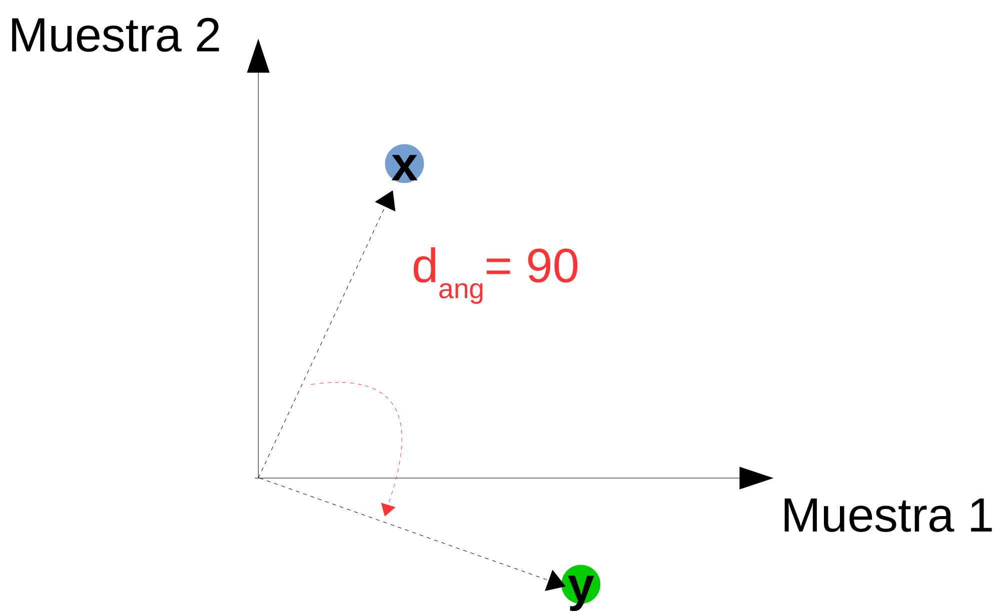
Similitud

Asumiendo que las señales están centradas:
“y” primero inicia el parte positiva, y luego se va a la negativa.
Si fuera una correlación negativa, “y” comenzaría en el lado contrario del eje de donde inicia “x”.
No podemos ver ninguna correlación entre las dos señales aquí.

$d_{ang} = 90$
$r = 0$







Correlación Cruzada Normalizada

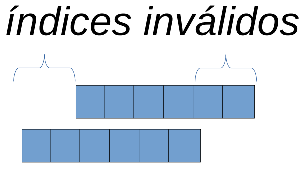
Capturar a “x” y “y” del micrófono 1 y 2, respectivamente.

Calcular *desfase máximo** en ambas direcciones (-max_d y max_d).

Definir “C”: arreglo del tamaño de todos los posibles desfases.

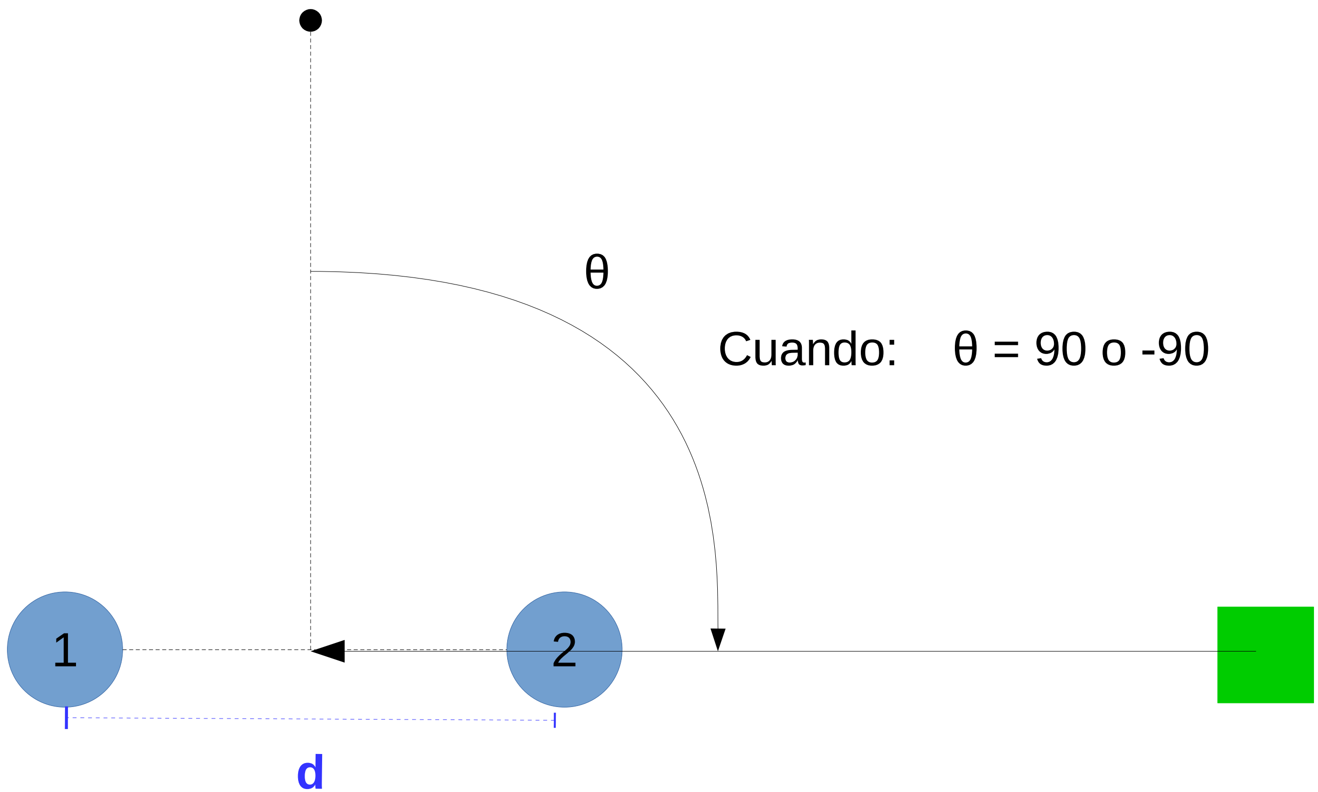
Para cada desfase “d” del rango [-max_d, max_d]:

- Desfasar a “y”, “d” muestras.
- Deshechar muestras de ambas señales que estén en índices invalidos:
- Calcular promedio y centrar muestras válidas
- Calcular Pearson para desfase “d” y guardar en “C”.



Resultado: “C” es el **Vector de Correlación Cruzada** (CCV).

*Desfase Máximo



Cuando: θ = 90 o -90

$$t_{2-1_{max}} = \frac{d}{V_{sound}}$$

Un Nuevo Amigo: octave

- Alternativa de código abierto a MATLAB.

  sudo apt-get install octave octave-signal

  - Útil para probar rapidamente operaciones de algebra lineal, sin utilizar bucles.
- Asistencia para hacer “prototipo” de ideas, antes de pasarlas a C/C++.
  - Si ya probamos que la idea es “sana”, podemos dedicarle el tiempo a implementarlo con JACK.

Y...

- Los **índices** de los arreglos manipulados en Octave van de 1 a N.
  - Recuerden que en C van de 0 a N-1.

- No, no hay una librería de JACK para octave.

Flojos...

También en Python

- Por si no les gusta Octave/MATLAB, y prefieren Python, en la página del curso estarán todos los siguientes ejercicios en Python.
- Bibliotecas necesarias:
  - numpy
  - matplotlib

Ejercicio #1

Vamos a crear dos señales, una desfasada puramente de la otra:

```
t = 1:100;
x = e.^(-t/10);
y = [zeros(1,15) x(1:end-15)];
```

Para ver las señales:

```
plot(x)
plot(y)
```

Ejercicio #1

Centramos señales:

```
x_c = x - mean(x);
y_c = y - mean(y);
```

Calculemos Pearson:

```
r = sum(x_c.*y_c)/(norm(x_c)*norm(y_c))
```

Desfasamiento en Octave/Matlab

Si quisiéramos desfasar a “y” por 5 muestras, uno pensaría hacerlo de la siguiente manera:

```
y_d = [zeros(1,5) y(1:end-5)];
```

Pero así estamos agregando ceros de una manera artificial, lo cual va a meter “ruido” en el cálculo de la correlación cruzada.

- Lo que debemos hacer es: desfasar a “y” quedándonos solo con las muestras válidas, y calcular la correlación utilizando solo las muestras correspondientes de “x”.

  En Matlab/Octave se hace así:

```
y_d = y(1:end-5);
x_d = x(6:end);
```

Ejercicio #1: desfase positivo

Vamos a desfasar a “y” por 5 muestras, y dejar solo las muestras válidas en ambas señales:

```
y_d = y(1:end-5);
x_d = x(6:end);
```

Centramos señales:

```
y_dc = y_d - mean(y_d);
x_dc = x_d - mean(x_d);
```

Calculemos Pearson:

```
r = sum(x_dc.*y_dc)/(norm(x_dc)*norm(y_dc))
```

Ejercicio #1: desfase negativo

Vamos a desfasar a “y” por -5 muestras, y dejar solo las muestras válidas en ambas señales:

```
y_d = y(6:end);
x_d = x(1:end-5);
```

Centramos señales:

```
y_dc = y_d - mean(y_d);
x_dc = x_d - mean(x_d);
```

Calculemos Pearson:

```
r = sum(x_dc.*y_dc)/(norm(x_dc)*norm(y_dc))
```

Ejercicio #1: desfase óptimo

Vamos a desfasar a “y” por -15 muestras, y dejar solo las muestras válidas en ambas señales:

```
y_d = y(16:end);
x_d = x(1:end-15);
```

Centramos señales:

```
y_dc = y_d - mean(y_d);
x_dc = x_d - mean(x_d);
```

Calculemos Pearson:

```
r_o = sum(x_dc.*y_dc)/(norm(x_dc)*norm(y_dc))
```

Consideraciones

- El caso de r = 1 o r = -1 es casi imposible en circunstancias reales:
  - Las señales capturadas en ambos micrófonos reciben audio de otros lados que no son la fuente.
    - Ruido
  - El desfase NO es la única diferencia que habrá entre las señales.
  - Por lo tanto, correlación perfecta es muy rara en ámbitos reales

Correlación Cruzada Normalizada en el Dominio de Frecuencia

- Interesantemente, lo siguiente aplica:

$$CCV = \frac{F^{-1}\left(X^{*} \cdot Y\right)}{\|\boldsymbol{x}\| \, \|\boldsymbol{y}\|}$$

Donde:

CCV = es el Vector de Correlación Cruzada
F^{-1} = la operación de transformada inversa de Fourier
$\cdot$ = Producto Punto a Punto
X, Y = las señales de “x” y “y” en el dominio de la frecuencia:

X = F(x)
Y = F(y)

* = el conjugado complejo (aplicado a la señal que se desfasa):

$X_i = a + ib$
$X_i^* = a - ib$

La **Correlación Cruzada** en el dominio del *tiempo*
equivale a
Producto entre conjugados complejos en el dominio de la *frecuencia*

Correlación Cruzada Normalizada en el Dominio de Frecuencia

- Interesantemente, lo siguiente aplica:

$$CCV = \frac{F^{-1}\left(X^{*} \cdot Y\right)}{\|\boldsymbol{x}\| \, \|\boldsymbol{y}\|}$$

Es una multiplicación *punto a punto.*

Se divide por las magnitudes porque es **normalizada**.

Donde:

CCV = es el Vector de Correlación Cruzada
F^{-1} = la operación de transformada inversa de Fourier
· = Producto Punto a Punto
X, Y = las señales de "x" y "y" en el dominio de la frecuencia:

X = F(x)
Y = F(y)

* = el conjugado complejo (aplicado a la señal que se desfasa):

$X_i = a + ib$
$X_i^* = a - ib$

La **Correlación Cruzada** en el dominio del *tiempo*
equivale a
Producto entre conjugados complejos en el dominio de la *frecuencia*

Índices-a-Desfases en el CCV (Octave)

- El resultado de la transformada inversa regresa al CCV de tal forma que los índices en Octave son:

  1 := correlación sin desfase.

  2 := correlación con desfase positivo 1.

  3 := correlación con desfase positivo 2.

- Y si, N := tamaño de señal:

  N := correlación con desfase negativo 1 (-1).

  N-1 := correlación con desfase negativo 2 (-2).

  N-2 := correlación con desfase negativo 3 (-3).

Índices-a-Desfases en el CCV (C)

- Los índices del CCV en C son:

    0 := correlación sin desfase.

    1 := correlación con desfase positivo 1.

    2 := correlación con desfase positivo 2.

- Y si, N := tamaño de señal:

    N-1 := correlación con desfase negativo 1 (-1).

    N-2 := correlación con desfase negativo 2 (-2).

    N-3 := correlación con desfase negativo 3 (-3).

Ejercicio #2

Obtenemos la transformada de Fourier de las señales centradas:

```
x_f = fft(x_c);
y_f = fft(y_c);
```

Calculemos el vector completo de un sólo paso:

```
ccv_f = conj(x_f).*y_f;
ccv = real(ifft(ccv_f))/(norm(x_c)*norm(y_c));
```

Ejercicio #2

Veamos al CCV:

plot(ccv)

¿Cuál es el desfase?

[m mi] = max(ccv)

m: máximo valor en ccv

mi: indice del máximo valor en ccv

Si mi < N/2, el desfase es:

desfase = mi-1

Si no:

desfase = mi-N-1

Ejercicio #3

¿Qué sucede si conjugamos la otra señal?

```
ccv_f = x_f.*conj(y_f);
ccv = real(ifft(ccv_f))/(norm(x_c)*norm(y_c));
```

Ejercicio #3

¿Qué sucede si conjugamos la otra señal?

```
ccv_f = x_f.*conj(y_f);
ccv = real(ifft(ccv_f))/(norm(x_c)*norm(y_c));
```

El CCV se “voltea”.

Ejercicio #3

Ahora, ¿cuál es el desfase?

[m mi] = max(ccv)

m: máximo valor en ccv

mi: indice del máximo valor en ccv

Si mi < N/2, el desfase es:

desfase = mi-1

Si no:

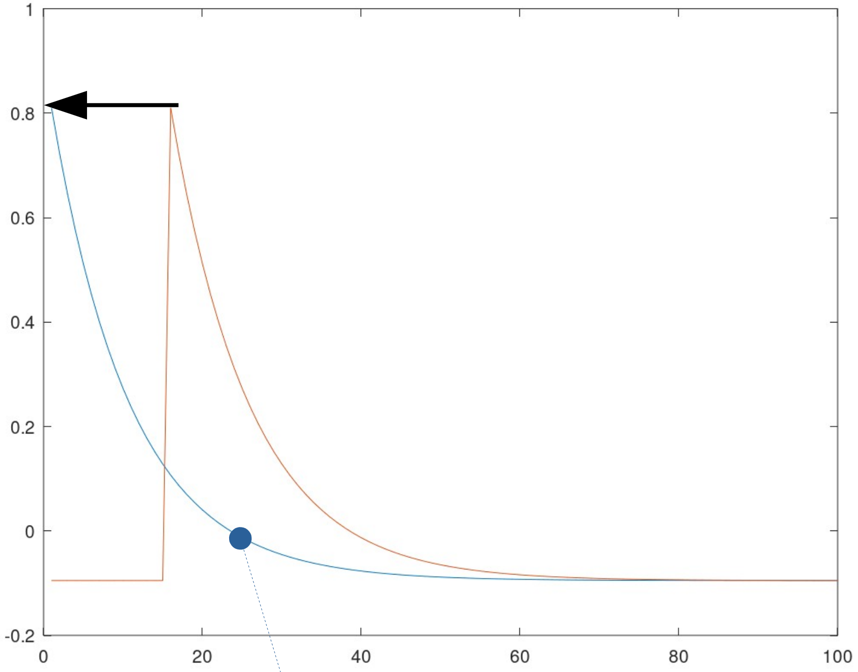
desfase = mi-N-1

Micrófono de Referencia

- Es el micrófono que no se “mueve”, y al que el otro micrófono trata de “acomodarse”.
  - Hasta el ejercicio #2, éste era “x”.
  - “y” era el que desfasábamos.

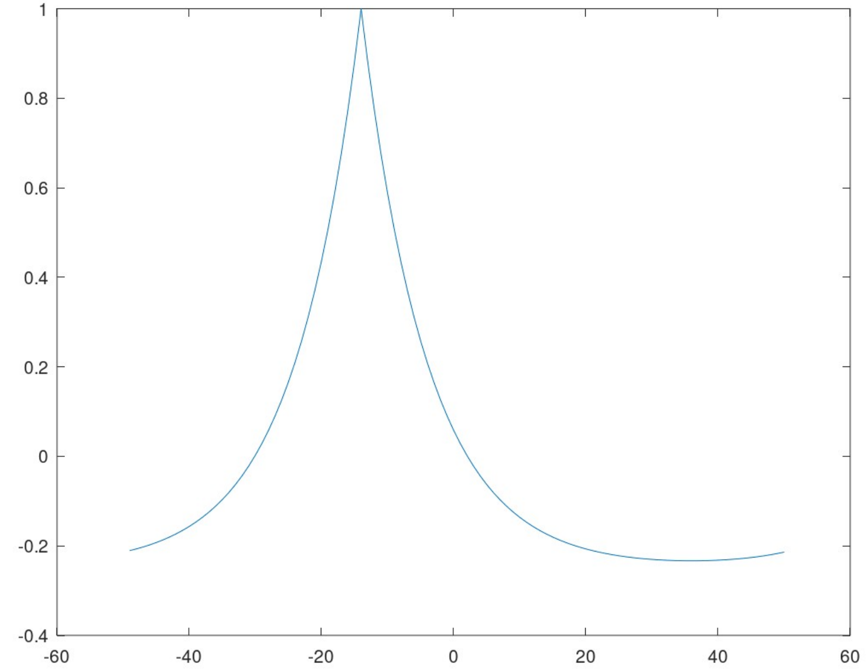
- En el caso del cálculo del CCV por medio de conjugar alguno de los dos micrófonos en el dominio de la frecuencia, podemos ver que:

ccv_f = x_f.*conj(y_f);

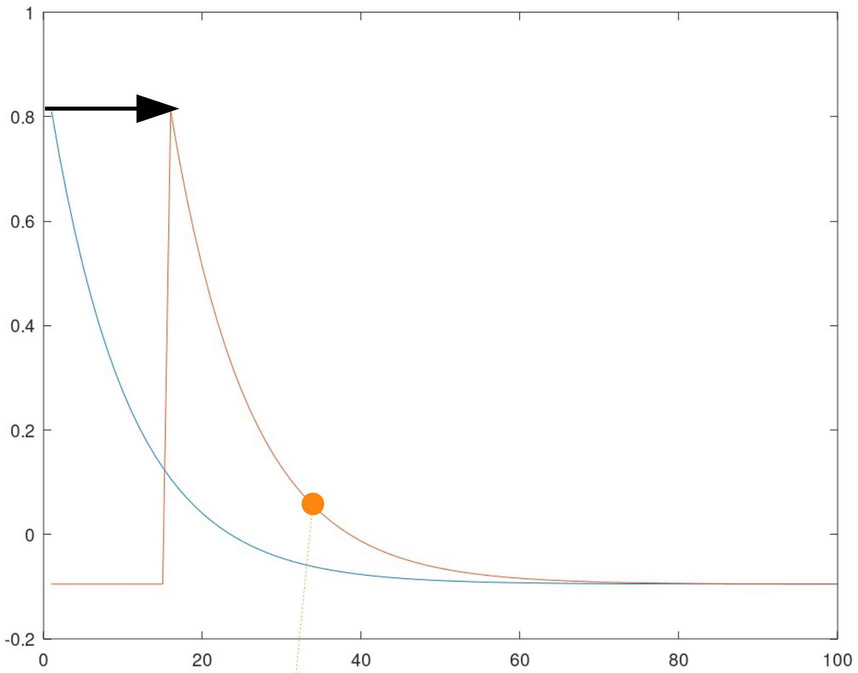


X es el micrófono de referencia

CCV
conjugando a **Y**

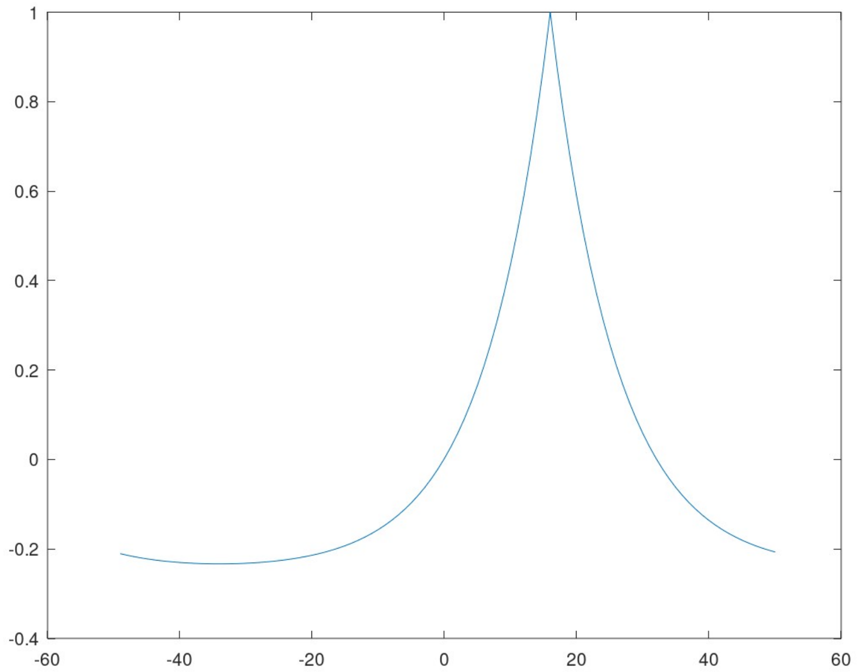


ccv_f = conj(x_f).*y_f;



Y es el micrófono de referencia

CCV
conjugando a **X**



El micrófono al que se le aplica el complejo conjugado es el micrófono que se “desfasa”.

El otro micrófono es el de referencia.

Beneficios del Cálculo del CCV en el Dominio de la Frecuencia

- Es más rápido, dada una implementación eficiente del FFT.
  - Entrega todo el CCV de un sólo golpe.
- Ya que estamos en el dominio de la frecuencia, podemos hacer “magia espectral”...

Correlación Cruzada Generalizada

- Interesantemente, lo siguiente aplica:

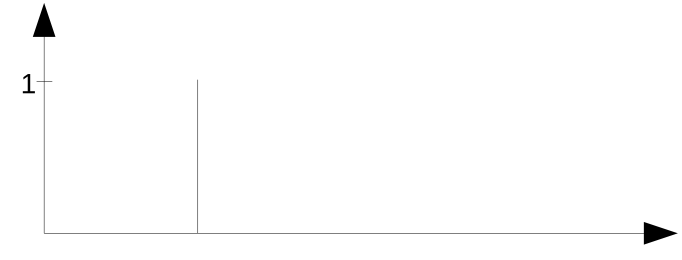
$$CCV_g = F^{-1}\left(\delta \cdot \left(X^* \cdot Y\right)\right)$$

Donde:

CCV_g = es el Vector de Correlación Cruzada Generalizada
δ = una función de filtrado espectral
F^{-1} = la operación de transformada inversa de Fourier
· = Producto Punto
X, Y = las señales de “x” y “y” en el dominio de la frecuencia:
X = F(x)
Y = F(y)
* = el conjugado complejo (aplicado a la señal que se desfasa):
$X_i = a + ib$
$X_i^* = a - ib$

Correlación Cruzada Generalizada

- La función de filtrado (δ) puede tomar varias formas dependiendo del problema que se quiere resolver.
- Un problema bastante visto es el que el “pico” dentro del CCV es muy “gordo”.
  - Lo cual es problemático en circunstancias con reverberación o con múltiples fuentes sonoras.
- Se prefieren picos “delgados”, o aproximaciones a funciones delta:



Correlación Cruzada Generalizada

- Sabemos que una función delta en el dominio del tiempo, es igual a un 1 en el dominio de la frecuencia.
- ¿No me creen?

```
a = zeros(1,64);
a(10) = 1;
figure(1); plot(a)
figure(2); plot(abs(fft(a)))
```

Correlación Cruzada Generalizada

- Por lo tanto, para forzar la presencia de funciones delta en el dominio del tiempo:
  - Podemos dividir todos los valores del dominio de la frecuencia por su propia magnitud.
  - Y así forzar que su magnitud siempre sea 1:

$$\frac{X^* \cdot Y}{\left\| X^* \cdot Y \right\|}$$

La Transformada de Fase

- Dada esa ecuación, podemos entonces definir a la función de filtrado espectral (δ) como:

$$\delta = \frac{1}{\left\| X^{*} \cdot Y \right\|}$$

- La cual es conocida como la Transformada de Fase.
  - Ya que al aplicarla, sólo queda la información de fase.
  - La información de magnitud resultante es siempre igual a 1 en todas las frecuencias

Correlación Cruzada Generalizada con la Transformada de Fase

$$CCV_{phat} = F^{-1}\left(\frac{X^{*} \cdot Y}{\left\| X^{*} \cdot Y \right\|}\right)$$

Donde:

CCV_{phat} = es el Vector de Correlación Cruzada Generalizada, a partir de la transformada de fase.
F^{-1} = la operación de transformada inversa de Fourier
$\cdot$ = Producto Punto
X, Y = las señales de “x” y “y” en el dominio de la frecuencia:
$X = F(x)$
$Y = F(y)$
* = el conjugado complejo (aplicado a la señal que se desfasa):
$X_i = a + ib$
$X_i^* = a - ib$
||.|| = es la magnitud del número complejo: $(a^2+b^2)^{1/2}$

Ejercicio #4

Aplicamos la transformada de fase:

```
ccv_fp = conj(x_f).*y_f./(abs(conj(x_f).*y_f));
```

Regresamos al dominio del tiempo

```
ccvp = real(ifft(ccv_fp));
```

Dividir entre las magnitudes de las señales centradas ya no es necesario en este caso:

Ya no hay magnitudes involucradas

Ejercicio #4

El desfase se cálcula de manera idéntica a los ejercicios 2 y 3.

Comparamos los dos ccv's creados:

```
plot(ccv)
plot(ccvp)
```

Consideraciones Importantes de GCC-PHAT

- Al aplicar PHAT, la IFFT se hace sensible a sangrado si ocurren discontinuidades entre el inicio y el fin de los valores de energía en ventanas pequeñas.
  - Lo cual resulta en valores grandes de correlación en el desfase 0, produciendo posibles confusiones.
- Para evitar esto, se debe aplicar la función Hann a la ventanas antes de hacer el FFT.

Ejercicio en C

- Escojan alguna de las formas descritas aquí para calcular la dirección de arribo.
- Impleméntenlo en JACK.
  - Si su proyecto sólo va a involucrar a una fuente:
    - Este ejercicio es parte de su proyecto final.
  - Si no, el siguiente tema será de su agrado.
- Pruébenlo con AIRA y el sistema de emulación (proveyendo dos canales).

Siguiente Clase

Estimación de Múltiples Direcciones de Arribo