

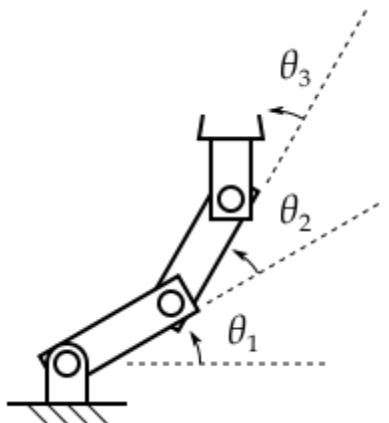
# FABRIK

Cinemática Inversa

Robótica  
computacional

Realizado por:  
Borja Barrera Villagrasa  
Jorge O. Blanchard Cruz

# Introducción



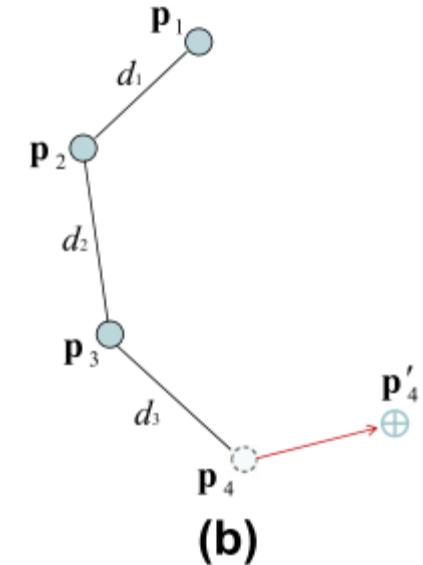
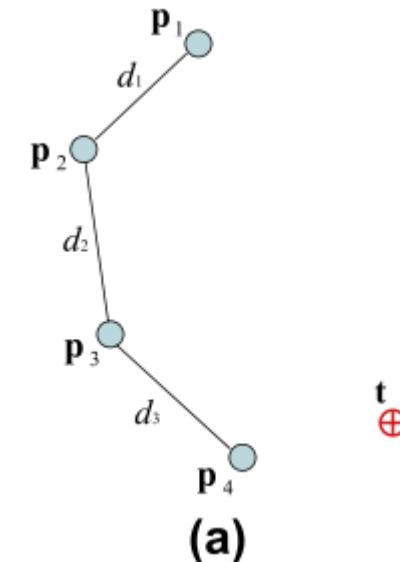
- FABRIK es un algoritmo de cinemática inversa utilizado en robótica y en motores gráficos.
- Método heurístico que utiliza las posiciones previamente calculadas de las articulaciones para buscar las actualizaciones en un modo iterativo hacia adelante y hacia atrás.
- Consiste en minimizar el error del sistema ajustando cada ángulo común a la vez.
- El método propuesto se inicia a partir de la última articulación de la cadena y trabaja hacia adelante, ajustando cada articulación a lo largo del camino. Después de eso, trabaja hacia atrás de la misma manera, con el fin de completar una iteración completa.
- Este método, en lugar de usar rotaciones de ángulo, trata de encontrar los lugares comunes como un problema de encontrar un punto en una línea; por lo tanto, se pueden ahorrar tiempo y cómputo.

# Algoritmo paso a paso (1/4)

1. **Comprobar si el objetivo es alcanzable;** Para ello se suman las distancias de los links de cada articulación. Si esta distancia es mayor o igual que la distancia que hay entre la primera articulación y el objetivo quiere decir que el objetivo es alcanzable.

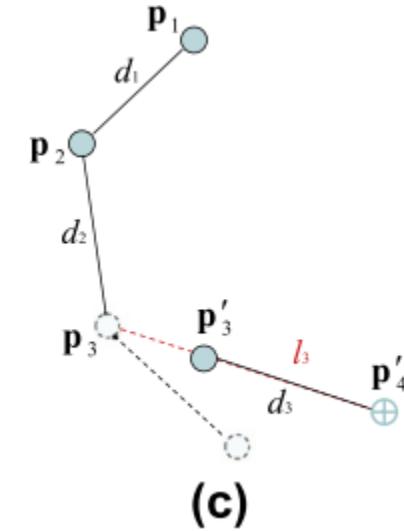
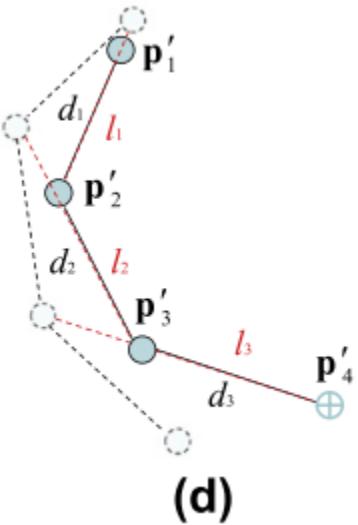
Si el objetivo está dentro de alcance, una iteración completa está constituida por dos etapas:

2. **En la primera etapa**, el algoritmo estima cada posición de la articulación a partir del efecto final  $\mathbf{p}_4$ :
  - i. Primero dejamos que el efecto final  $\mathbf{p}_4$  se coloque en la posición del objetivo  $\mathbf{t}$  (siendo luego  $\mathbf{p}'_4$ ). (figura (b)).



# Algoritmo paso a paso (2/4)

- II. Se traza una línea que pasa a través de la articulación anterior  $p_3$  y la del efector final  $p'_4$ .
- III. Se coloca la articulación anterior  $p_3$  sobre la línea trazada anteriormente a la distancia de su link  $l_3$  (quedando  $p'_3$ ).
- IV. Se repite el proceso hasta que se han colocado todas las articulaciones (**figura (d)**).

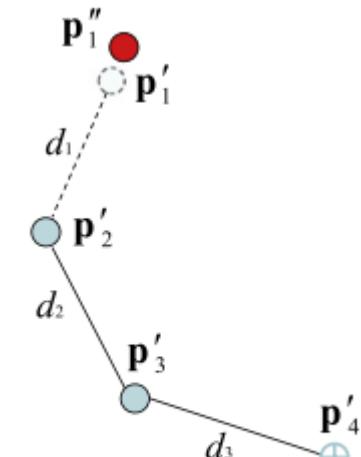


# Algoritmo paso a paso (3/4)

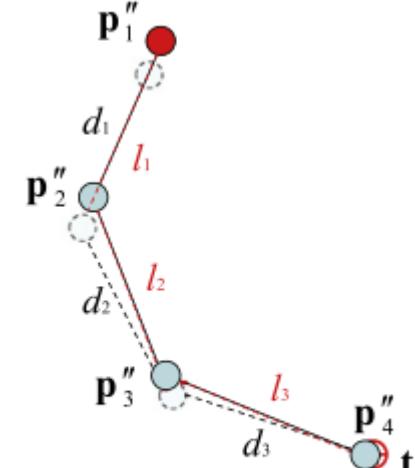
3. La articulación base no debe ser diferente de su posición inicial. Por tanto, en la **segunda etapa** el algoritmo recoloca la articulación base a su origen recalculando las posiciones de las demás articulaciones.

El procedimiento es exactamente el mismo que el la primera etapa, pero en sentido contrario.

- i. Se coloca la articulación base  $p'_1$  en su posición inicial  $p''_1$  (**quedando  $p''_1$** ) (figura (e)).
- ii. Se traza una línea que pasa a través de la articulación base  $p''_1$  y la siguiente articulación  $p'_2$ .
- iii. Se coloca la articulación siguiente  $p'_2$  sobre la línea trazada a la distancia de su link  $l_1$  (**quedando  $p''_2$** ).
- iv. Se repite el proceso hasta que se han recolocado todas las articulaciones.



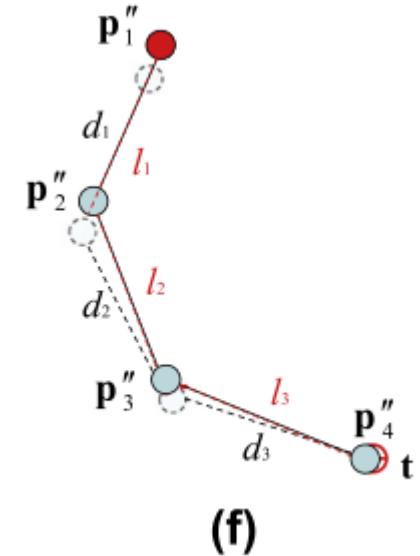
(e)



(f)

# Algoritmo paso a paso (4/4)

- Esto completa una sola iteración. Se puede apreciar que el efecto final  $p''_4$  se ha desplazado del objetivo  $t$ . Para corregir esto se ejecuta el algoritmo desde el paso 2 hasta que el efecto final  $p''_4$  queda sobre dicho objetivo  $t$  o cerca de este con un mínimo fallo.



# THE END!

¡Gracias por su atención  
y qué tengan un buen día!