

FABRIK

- Es un nuevo método heurístico que utiliza las posiciones previamente calculadas de las articulaciones para buscar las actualizaciones en un modo iterativo hacia adelante y hacia atrás.
- FABRIK 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 delante, 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.

Una iteración completa del algoritmo FABRIK

Input: The joint positions \mathbf{p}_i for $i = 1, \dots, n$, the target position \mathbf{t} and the distances between each joint $d_i = |\mathbf{p}_{i+1} - \mathbf{p}_i|$ for $i = 1, \dots, n-1$

Output: The new joint positions \mathbf{p}_i for $i = 1, \dots, n$.

```
1.1 % The distance between root and target
1.2 dist = |p1 - t|
1.3 % Check whether the target is within reach
1.4 if dist > d1 + d2 + ... + dn-1 then
1.5     % The target is unreachable
1.6     for i = 1, ..., n-1 do
1.7         % Find the nearest point on sphere, with centre the joint position pi and radius the distance di, from a
           point is space, t
1.8         pi+1 = NearestPointSphere(pi, di, t);
1.9     end
1.10 else
1.11     % The target is reachable; thus, set as b the initial position of the joint p1
1.12     b = p1
1.13     % Check whether the distance between the end effector pn and the target t is greater than a tolerance.
       dif A = |pn - t|
1.14     while dif A > tol do
1.15         % STAGE 1: FORWARD REACHING
1.16         % Set the end effector pn as target t
1.17         pn = t
1.18         for i = n-1, ..., 1 do
1.19             % Find the nearest point on sphere, with centre the joint position pi+1 and radius the distance
               di, from a point is space, pi
1.20             pi = NearestPointSphere(pi+1, di, pi);
1.21         end
1.22         % STAGE 2: BACKWARD REACHING
1.23         % Set the root p1 its initial position.
1.24         p1 = b
1.25         for i = 1, ..., n-1 do
1.26             % Find the nearest point on sphere, with centre the joint position pi and radius the distance
               di, from a point is space, pi+1
```

```

                                pi 1.28 = NearestPointSphere(pi , di , pi+1);
1.29                                end
                                diff A = |pn 1.30 - t|
1.31                                end
1.32 end
1.33
1.34 % Where the function NearestPointSphere(X,Y,Z), finds the nearest point on a sphere
from a point in space. X is the sphere's centre, Y is the sphere's radii and Z is the point
in space.

```

Por lo tanto, asumimos que p_1, \dots, p_n son las posiciones conjuntas de un manipulador. Para el caso simple donde sólo existe un único efector final, tome p_1 como la articulación de la raíz y p_n como el efector final.

El objetivo es t , y la posición de base inicial es b .

En primer lugar calcular las distancias entre cada articulación

$$d_i = |p_{i+1} - p_i| \text{ for } i = 1, \dots, n-1.$$

Entonces, para comprobar si el objetivo es alcanzable o no, encontramos la distancia entre la raíz y el objetivo, **dist**, y si esta distancia es menor que la suma total de todas las distancias entre conjuntos, $dis < \sum_{i=1}^{n-1} d_i$ el objetivo está a nuestro alcance; de lo contrario, es inalcanzable.

Si el objetivo está dentro de alcance, una iteración completa está constituida por dos etapas. En la primera etapa, el algoritmo estima cada posición de la articulación a partir del efector final, p_n , moviéndose hacia el interior, a la base del manipulador, p_1 . Por lo tanto, dejamos que la nueva posición del efector final sea la posición de destino, $p'_n = t$.

La nueva posición conjunta de la (n-1) -ésimo, p'_{n-1} , se asigna como el punto de la esfera Σ_{n-1} más cercano, con el centro de la posición conjunta y radios p'_n , la distancia d_{n-1} de la posición conjunta p_{n-1} .

Del mismo modo, la nueva posición conjunta de la (n - 2) -ésimo, p'_{n-2} , se selecciona como el punto sobre la esfera Σ_{n-2} más cercana, con el centro de la articulación de la posición p'_{n-1} y radios de la distancia d_{n-2} de la articulación p_{n-2} .

El algoritmo continúa hasta que se calculan todas las nuevas posiciones comunes, incluyendo la raíz, p'_1 . El punto en una esfera de un punto en el espacio más cercano se encuentra simplemente tomando un punto a lo largo de la línea que une el centro de la esfera hasta el punto, que tiene distancia desde el centro igual al radio de la esfera.

Una iteración completa finaliza cuando se repite el mismo procedimiento, pero esta vez a partir de la articulación de la raíz y en movimiento hacia el exterior hasta el efector terminal. Por lo tanto, dejamos que la nueva posición de la primera articulación, p''_1 , sea su posición inicial b . Entonces, la nueva posición conjunta p''_2 se asigna como el punto más cercano en la esfera Σ_1 , con el centro de p'_2 y radios la distancia conjunta d_1 de la p'_2 . Este procedimiento se repite para todas las articulaciones restantes, incluyendo el efector de extremo.