Calculate the Point of the Slave Mechanism

ISSUES

A slave mechanism is attached on two arbitrary point at a torsional hinge `Hinge A` and spherical hinge `Hinge B` thougth `SlaveArm A` and `SlaveArm B` respectively, which means that `SlaveArm A` can only rotate at a certain angle to the `Hinge A` while there is no this limitations on `SlaveArm B`.

Therefore, the intersection node `SlavePoint` of `SlaveArm A` and `SlaveArm B` can be determined if the length ($LengthSlave_a$ and $LengthSlave_b$) and attached location ($AttachPoint_a^{g/l}$ and $AttachPoint_b^{g/l}$) of two arms and the rotation plane of `Hinge A` ($RotaVec_a^{g/l}$) are given, where the prime notion indicates it is a *global* or *local* coordinate.

PROCESS

Equation Setting Up

Local system can be used to describe the motion of `SlaveArm A`. As the `SlaveArm A` is rotating in a certain plane with the normal vector $RotaVec_a^l$, the motion curve at the local system of `Hinge A` is a circle where the center is $AttachPoint_a^l$ and the radius is $LengthSlave_a$. According to this reference, the parametrical function of this curve, which is also the local coordinate of $SlavePoint$, can be put as:

$$circle(\theta) = SlavePoint^l(\theta) = AttachPoint_a^l + LengthSlave_a \cdot cos(\theta) \cdot \vec{a} + LengthSlave_b \cdot sin(\theta) \cdot \vec{b}$$

where $\vec{a}$ and $\vec{b}$ are two vectors in the rotation plane which are normal to each others.

Meanwhile, the length constraints of `SlaveArm B` can be obtained:

$$eqn1 : \|SlavePoint^g - AttachPoint_b^g\| - lb == 0$$

Linear Transformation

One more thing should be paid attention is the tansformation of global and local system. As known, here is only about the linear transforming including ratation and translation but nothing with scale. So if the local coordinate is replaced by global one, it also stands for a same shape.

For the convenience, all the $SlavePoint$ should be transformed into a consistent formation, and the circle curve function can be rewriten as a global formation:

$$SlavePoint^g(\theta) = M_{SP} \cdot SlavePoint^l(\theta) + O_L$$

where M_{SP} is the rotation matrix and the O_L is the local origin.

Equation Solving

Adopting the circle curve function to the `eqn1`, the result is:

$$\|M_{SP} \cdot AttachPoint_a^l + LengthSlave_a \cdot cos(\theta) \cdot M_{SP} \cdot \vec{a} + LengthSlave_a \cdot sin(\theta) \cdot M_{SP} \cdot \vec{b} + O_L - AttachPoint_b^g\| - LengthSlave_b == 0$$

To be more simplifier, it can be rewriten as

$$A + B \cdot cos^2(\theta) + C \cdot sin^2(\theta) + D \cdot cos(\theta) + E \cdot sin(\theta) + F \cdot cos(\theta) \cdot sin(\theta) == 0$$

where:

$A = \|M_{SP} \cdot AttachPoint_a^l + O_L - AttachPoint_b^g\|^2 - LengthSlave_b^2$

$B = LengthSlave_a^2 \cdot \|M_{SP} \cdot \vec{a}\|^2$

$C = LengthSlave_a^2 \cdot \|M_{SP} \cdot \vec{b}\|^2$

$D = 2 \cdot LengthSlave_a \cdot (M_{SP} \cdot AttachPoint_a^l + O_L - AttachPoint_b^g)' \cdot M_{SP} \cdot \vec{a}$

$E = 2 \cdot LengthSlave_a \cdot (M_{SP} \cdot AttachPoint_a^l + O_L - AttachPoint_b^g)' \cdot M_{SP} \cdot \vec{b}$

$F = 2 \cdot LengthSlave_a \cdot LengthSlave_a \cdot M_{SP} \cdot \vec{a} \cdot M_{SP} \cdot \vec{b}$

With the help of `MATLAB`, the result of this equation is $\theta = 2 * arctan(\alpha)$ where α should meet these conditions:

$$eqn.2 : A + B + D + 2E \cdot \alpha + 2F \cdot \alpha + 2A \cdot \alpha^2 + A \cdot \alpha^4 + B \cdot \alpha^4 + 4C \cdot \alpha^2 + 2E \cdot \alpha^3 == D \cdot \alpha^4 + 2F \cdot \alpha^3 + 2B \cdot \alpha^2$$

Transfroming to a standard formation, $eqn.2$ can be put as:

$$eqn.2.1 : M \cdot \alpha^4 + N \cdot \alpha^3 + P \cdot \alpha^2 + Q \cdot \alpha^1 + T == 0$$

where:

$M = A + B - D$

$N = 2E - 2F$

$P = 2A + 4C - 2B$

$Q = 2E + 2F$

$T = A + B + D$

Solution for Univariate Quartic Equation

There is a standard methed to deal with the `univariate quartic equation` such as $eqn2.1$ and here are several situations:

- if **M=0**, eqn2.1 can be rewriten as a `univariate cubic equation` formation, but:
  - if **N=0**, it continues to be a `univariate quadratic equation` formation, but:
    - if **P=0**, it turns to be a `linear equation`, but:
      - if **Q=0**, but:
        - if **T≠0**, there is **no solution**
        - if **T=0**, there are **too many solution**
      - if **Q≠0**, there is a unique solution, $\alpha = -Q/T$
    - if **P≠0**, the solution for `univariate quadratic equation` depends on the value of parameters:
      - if $Q^2 - 4 \cdot P \cdot T < 0$, there is **no solution**
      - if $Q^2 - 4 \cdot P \cdot T \geqq 0$, $\alpha = \frac{-Q \pm \sqrt{Q^2 - 4 \cdot P \cdot T}}{2 \cdot P}$
  - if **N≠0**, according to the `ShengJin Method`, the algorithm for `univariate cubic equation` is easy to get.
- if **M≠0**, according to the `TianHeng Method`, the algorithm for `univariate quartic equation` is easy to get.

After solved the $eqn2.1$, the value of α can be put in to the equation $\theta = 2 * arctan(\alpha)$ and the value of θ is got.