Unconstrained Inverse Kinetics · Week I

CCO · Constraint Continous Optimization

Wednesday 2nd September, 2009

Johan Sejr Brinch Nielsen

Email:	zerrez@diku.dk
Cpr.:	260886-2547

Dept. of Computer Science,
University of Copenhagen

A: End-Effector Function

A homegenous transformation works by multiplying n matrices (one for each connecting angle) with a point, like so: $T_0T_1T_2 \dots T_n \cdot p$
Where the ith tranformation matrix T_i represents the ith angle and p is the point that needs transformation.

Each matrix will rotate the intermediate point with matrices associated angle and tanslate the point with the associated length.

The point p needs to be expanded with an extra "dimension" to make the math work (watch out; it's a trick).

The transformation function f now becomes:
$f(\theta^n) = T_1T_2T_3 \dots T_n \cdot x^d$
Where θ^n is the n dimensional angle vector and x is the d dimensional point to transform. If the cartesian coordinates were $(1, 2, 3)$ then the homegenous coordiate becomes $(1, 2, 3, 1)$.

In our particular case (with just 2 dimensions and 3 angles) the transformation function f becomes:
$f(\theta^3) = T_1T_2T_3 \cdot x^2$
Where x is the 2-dimensional point.

In 2 dimensions the transformation matrix i becomes:

$$\begin{vmatrix} \cos\theta_i & -\sin\theta_i & x_i \\ \sin\theta_i & \cos\theta_i & y_i \\ 0 & 0 & 1 \end{vmatrix}$$

B: Derivate of the End-Effector

The End-Effector is the resulting point after the transformation described above. This can be expressed as a vector parameterized by the angles by multiplying the starting point to the transformation matrices. The task is to get the end effector as close the goal as possible.

The resulting term describing the end effector will consist of addition of sin / cos and constant terms. The derivative can be easily computed by computing this for each of the different variables (the angles).

C: Root Search Problem

The initial guess θ^3 yields the first end effector by:

$$e = f(\theta^3)$$

Denote the difference between the goal g and the end effector e by $\Delta\theta$. The goal is now to find the root of $\Delta\theta$, hence the problem is that of a root-search.

D: Nonlinear Newton Method

We need to find a good guess on $\Delta\theta$ such that:

$g' = f(\theta + \Delta\theta)$ is closer to g than $f(\theta)$.

A simple guess is to look at the first Tayler expansion:

$$g' = f(\theta_0) + \frac{\delta f(\theta_0)}{\delta\theta_0}\Delta\theta$$

With a rest of $O(|| \Delta\theta^2 ||)$.

Letting,

$$\theta_{i+1} = \theta_i + \frac{\delta f(\theta_i)}{\delta\theta_i}\Delta\theta_i$$

should yield a better guess than the θ_i. This method approximates the goal in a linear way, always leaving a rest. Hopefully the rest, hence the distance to the goal, will close in on zero.

Pseudo Code

```
ik-solver(t0, g, e)
 t = t0
 e = f(t)
 while( |g-e| > e) do
  dt = j_0^-1(g-e)
  t  = t + dt
  e  = f(t)
 end while
 return dt
end
```