Unconstrained Inverse Kinematics as The Solution of Nonlinear Equations

Kenny Erleben[1]

[1]Department of Computer Science
University of Copenhagen

2009

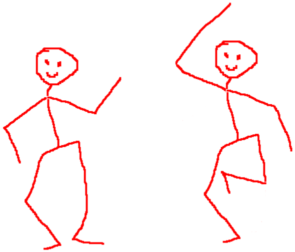
What does it mean?

Kinematics The "study" of motion without forces and masses.

Inverse From a known starting- and ending position try to figure out how to make the motion inbetween

Why is Inverse Kinematic (IK) Difficult?

Well let us look at happy man



What can we tell about the motion of a happy man?

Happy Man

- Motion is highly non-linear (both spatial and temporal)
- Motion is high dimensional (many joints)
- Motion is discontineous

So we are dealing with

High dimension non-linear non-smooth equations

Yrk! This is going to take some effort to compute

Are there other Problems?

Looking at the problem of finding a solution for the motion

- Loss of degree of freedom (ill-conditioness, singularity)
- More than one motion to get to the end position (uniqueness)
- End position might be unreachanle (existence)

Yikes we are **NOT** guaranteed

A well conditioned problem with one unique solution

We must expect a lot of numerical mess

End-Effector Position

We have a set of joint parameters, $\vec{\theta}$ we can tweak and change and gain explicit control over the end-effectors position and orientation, $\vec{e}$.

Given a serial mechanism we can set up a coordinate transformations from one joint frame to the next. Thus we can find one transformation that takes a point specified in the frame of the end-effektor into the root frame. We write it in a general way as

$$\vec{e} = \vec{f}(\vec{\theta}) \tag{1}$$

A 2D Serial Chain Robot Example

Simplifying Assumptions:

- Just position of origo of end-effector, we ignore orientation
- Only revolute joints, ith joint has joint angle θ_i.
- Fixed size rigid links given by the vector $\vec{t}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^T$
- Using Homegeneous Coordinates

So let us write up our $\vec{e} = \vec{f}(\vec{\theta})$

$$\vec{f}(\vec{\theta}) = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & x_1 \\ -\sin\theta_1 & \sin\theta_1 & y_1 \\ 0 & 0 & 1 \end{bmatrix} \cdots \begin{bmatrix} \cos\theta_i & \sin\theta_i & x_i \\ -\sin\theta_i & \sin\theta_i & y_i \\ 0 & 0 & 1 \end{bmatrix} \cdots \begin{bmatrix} \cos\theta_n & \sin\theta_n & x_n \\ -\sin\theta_n & \sin\theta_n & y_n \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{2}$$

Simplifying Notation

Let us write

$$\mathbf{T}_i = \begin{bmatrix} \cos\theta_i & \sin\theta_i & x_i \\ -\sin\theta_i & \cos\theta_i & y_i \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

Then

$$\vec{f}(\vec{\theta}) = \mathbf{T}_1 \mathbf{T}_2 \cdots \mathbf{T}_n \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{4}$$

Introducing $\Delta\vec{\theta}$

Initially we know the value of $\vec{\theta} = \vec{\theta}_0$ and a desired goal state for the end-effector, $\vec{g}$. The corresponding initial state of the end-effector is given by

$$\vec{e}_0 = \vec{f}(\vec{\theta}_0) \tag{5}$$

Writting

$$\vec{\theta} = \vec{\theta}_0 + \Delta\vec{\theta} \tag{6}$$

Our task is now to compute $\Delta\vec{\theta}$ such that

$$\vec{g} = \vec{f}(\vec{\theta}_0 + \Delta\vec{\theta}) \tag{7}$$

Taylor Series Expansion

Next we perform a Taylor series expansion of the RHS

$$\vec{g} = \vec{f}(\vec{\theta}_0) + \frac{\partial \vec{f}(\vec{\theta}_0)}{\partial \vec{\theta}} \Delta\vec{\theta} + O(||\Delta\vec{\theta}||^2) \tag{8}$$

We introduce the notation

$$\mathbf{J}_0 = \frac{\partial \vec{f}(\vec{\theta}_0)}{\partial \vec{\theta}} \tag{9}$$

And call this matrix the Jacobian. Next we throw away the remainder term of the Taylor series expansion, to obtain the approximation

$$\vec{g} = \vec{f}(\vec{\theta}_0) + \mathbf{J}_0 \Delta\vec{\theta} \tag{10}$$

Isolate the Unknowns

Recall that $\vec{e}_0 = \vec{f}(\theta_0)$ and assume that $\mathbf{J}_0$ is invertible then

$$\Delta\vec{\theta} = \mathbf{J}_0^{-1}(\vec{g} - \vec{e}_0) \tag{11}$$

This is a linear model for taking us as as close to $\vec{g}$ as we can with a linear step. Thus we may not get to $\vec{g}$ in one step. To solve this we will keep on take more steps until we get close enough.

The Jacobian Inverse Method

Algorithm ik-solver($\vec{\theta}_0, \vec{g}, \varepsilon$)
  $\vec{\theta} = \vec{\theta}_0$
  $\vec{e} = \vec{f}(\vec{\theta})$
  while($|\vec{g} - \vec{e}| > \varepsilon$ do
    $\Delta\vec{\theta} = \mathbf{J}_{\theta}^{-1}(\vec{g} - \vec{e})$
    $\vec{\theta} = \vec{\theta} + \Delta\vec{\theta}$
    $\vec{e} = \vec{f}(\vec{\theta})$
  end while
  return $\Delta\vec{\theta}$
End Algorithm

This is a non-linear Newton Method.

Only Two Problems

We only ned to know

- how to compute $\vec{f}(\vec{\theta})$?
- how to compute $\mathbf{J}^{-1}$?

Non-linear Newton Method

Theory: If $\vec{f}$ is contineously differentiable then `ik-solver` will

- Have quadratic convergence if $\vec{\theta}_0$ is close enoguh to the solution otherwise we may get linear convergence.
- Guarantee to find a solution to vector equation $\vec{g} = \vec{f}(\vec{\theta})$.

So we like the Newton Method.

The Newton Step

In our IK-solver we need to invert the Jacobian matrix in the computation

$$\Delta\vec{\theta} = \mathbf{J}_{\theta}^{-1}(\vec{g} - \vec{e}) \tag{12}$$

Note a new Jacobian is computed in every iteration of the Newton Method. The Jacobian is

- Most likely non-square since we have more joint parameters than degrees of freedom of the end-effector
- Most likely it is ill-posed or even singular (rank deficient) due to special alignment of motion.

So we can not simply invert it:-(

Inverting the Jacobian

- We could use the pseudo-inverse
- We could use singular-value decomposition
- We could simply use the jacobian transpose as an approximation
- Other matrix factorizations, iterative methods etc..

Computing The Jacobian

Recall

$$\vec{f}(\vec{\theta}) = \mathbf{T}_1 \mathbf{T}_2 \cdots \mathbf{T}_n \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{13}$$

And that

$$\mathbf{J}_{ij} = \frac{\partial \vec{f}_i(\theta)}{\partial \vec{\theta}_j} \tag{14}$$

Computing The Jacobian

So

$$\frac{\partial \vec{f}(\vec{\theta})}{\partial \vec{\theta}_i} = \frac{\partial}{\partial \vec{\theta}_i}\left(\mathbf{T}_1 \cdots \mathbf{T}_i \cdots \mathbf{T}_n \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right) = \mathbf{T}_1 \cdots \frac{\partial \mathbf{T}_i}{\partial \vec{\theta}_i} \cdots \mathbf{T}_n \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{15}$$

And straightforward differentiation yields

$$\frac{\partial \mathbf{T}_i}{\partial \vec{\theta}_i} = \frac{\partial}{\partial \vec{\theta}_i} \begin{bmatrix} \cos\theta_i & -\sin\theta_i & a_i \\ \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -\sin\theta_i & -\cos\theta_i & 0 \\ \cos\theta_i & -\sin\theta_i & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{16}$$

Making it Effective

Looking at

$$\partial_i \vec{f}(\vec{\theta}) = \underbrace{\mathbf{T}_1 \cdots \mathbf{T}_{i-1}}_{\mathbf{A}_i} \partial_i \mathbf{T}_i \underbrace{\mathbf{T}_{i+1} \cdots \mathbf{T}_n}_{\mathbf{B}_i} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{17}$$

We notice that

$$\mathbf{A}_{i+1} = \mathbf{A}_i \mathbf{T}_i \tag{18}$$

$$\mathbf{B}_i = \mathbf{T}_i \mathbf{B}_{i+1} \tag{19}$$

By pre-computation of **A** and **B** terms we can speed-up our computations.

A Geometric Interpretation

Let $\left[\vec{e}\right]_n^T = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ then

$$\partial_i \vec{f}(\vec{\theta}) = \mathbf{A}_i \partial_i \mathbf{T}_i \mathbf{B}_i \left[\vec{e}\right]_n \tag{20}$$

- $\mathbf{B}_i$ simply transforms the end-effector into the i^{th} joint frame.

$$\left[\vec{e}\right]_i = \mathbf{B}_i \left[\vec{e}\right]_n \tag{21}$$

- Multiplication by $\partial_i \mathbf{T}_i$ gives the end-effector position change wrt. $\vec{\theta}_i$ in the i^{th} frame,

$$\left[\partial_i \vec{e}\right]_i \partial_i \mathbf{T}_i \left[\vec{e}\right]_i \tag{22}$$

- Finally, $\mathbf{A}_i$ will transform the end-effector changes $\left[\partial_i \vec{e}\right]_i$ from the i^{th} frame into the root frame.

Reusing the $\vec{f}$-function

So $\vec{e} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ then we may abstactly write

$$\left[\vec{e}\right]_0 = \vec{f}(\theta_1..\theta_n, \left[\vec{e}\right]_n) \tag{23}$$

Now we compute

$$\left[\vec{e}\right]_i = \vec{f}(\theta_{i+1}..\theta_n, \left[\vec{e}\right]_n) \tag{24}$$

Next we compute

$$\mathbf{J}_{\cdot,i} = \vec{f}(\theta_1..\theta_{i-1}, \partial_i \mathbf{T}_i \left[\vec{e}\right]_i) \tag{25}$$