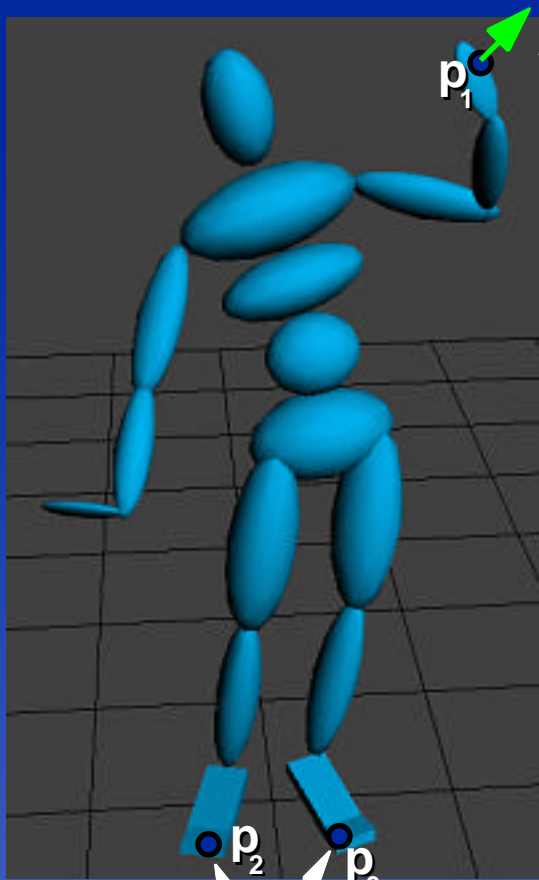


# Inverse Kinematics

*Sebastian Grassia*  
School of Computer Science



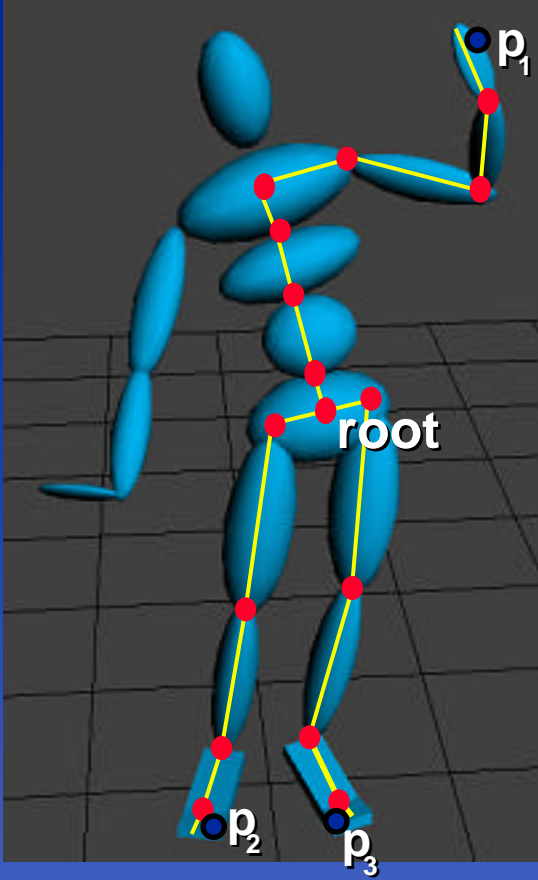
# Define the Problem



## Goals


- Drag in realtime with mouse
- Maintain multiple constraints
- Figure responds in “reasonable” ways

# ... Then Name the System!



## Use standard transformation hierarchy

$$p_1 = T(x, y, z)R(\theta, \phi, \varphi)TR(\gamma) \dots TR(\sigma) p_b$$

$q(t) =$  

$$p_1 = C(q(t))$$

# Idea #1

Let

$p_m(t)$  = 3D mouse positioner at time  $t$

$C(q(t))$  = position of hand at time  $t$

... then solve for  $q$  such that

$$C(q(t)) = p_m(t)$$

# Unfortunately...

$C(q(t)) = p_m(t)$  contains rotations, therefore nonlinear,  
and very difficult to solve

## Can be solved for:

- articulated chains

## Has real problems with:

- branching hierarchies
- multiple constraints
- convincing motion

## Idea #2

Let's differentiate!

$$\frac{\partial}{\partial t}(p_m(t) = C(q(t)))$$



$$\dot{p}_m = \frac{\partial C}{\partial q} \dot{q}$$

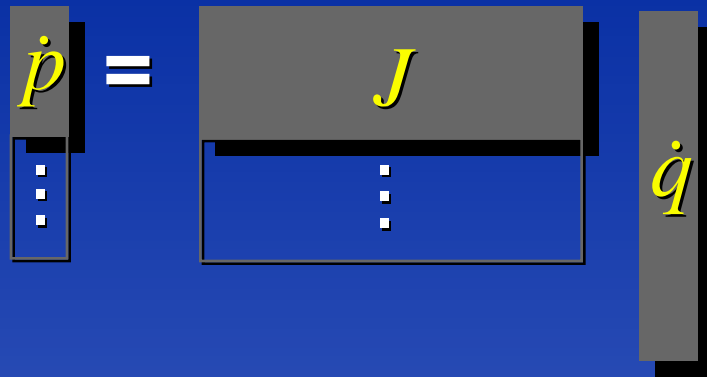
desired – actual

Jacobian  $J$

state space velocity

# A Diffy Q by any other name...

$$\dot{p} = J\dot{q}$$



The diagram illustrates the equation  $\dot{p} = J\dot{q}$  using matrix representations. On the left, a vertical grey rectangle contains the symbol  $\dot{p}$  in yellow, with a white equals sign to its right. Below this, a blue rectangle contains three white dots, representing a column vector. In the center, a grey rectangle contains the symbol  $J$  in yellow. Below this, a blue rectangle contains three white dots, representing a matrix. On the right, a vertical grey rectangle contains the symbol  $\dot{q}$  in yellow, representing a column vector.

Underdetermined linear system  
Can solve for  $\dot{q}$

Iterate during interaction using:

$$q(t + \Delta t) = q(t) + \Delta t \dot{q}(t)$$

# Are we there yet?

Alas, no.

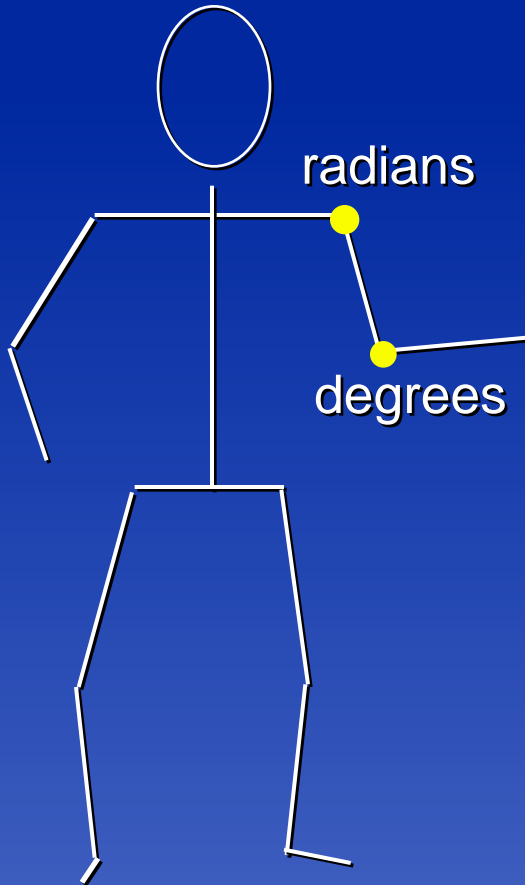
We got a solution for  $\dot{q}$

but it may not be a *reasonable* one.

## Why not?



# Scaling



When there's a choice,  
least squares solver always  
moves shoulder before elbow

Measuring both angles in radians  
still leaves figure stiff-armed

**What can we do about this?**

# Let's get physical!

## Aristotlean physics:

$$f = Mv$$

*Why not  $f = Ma$  ?*

$$f_G + f_C = M\dot{q}$$

Generalized Force

Mass Matrix

Constraint Force

# Blend on High for 2 minutes...

## Recall

Constraints

$$p = C(q)$$

Solve

$$\dot{p} = J\dot{q}$$

Where

$$J = \frac{\partial C}{\partial q}$$

**Define:**  $W = M^{-1}$

$$f_G + f_C = M\dot{q}$$

Lagrange Multipliers

$$f_G + J^T \lambda = M\dot{q}$$

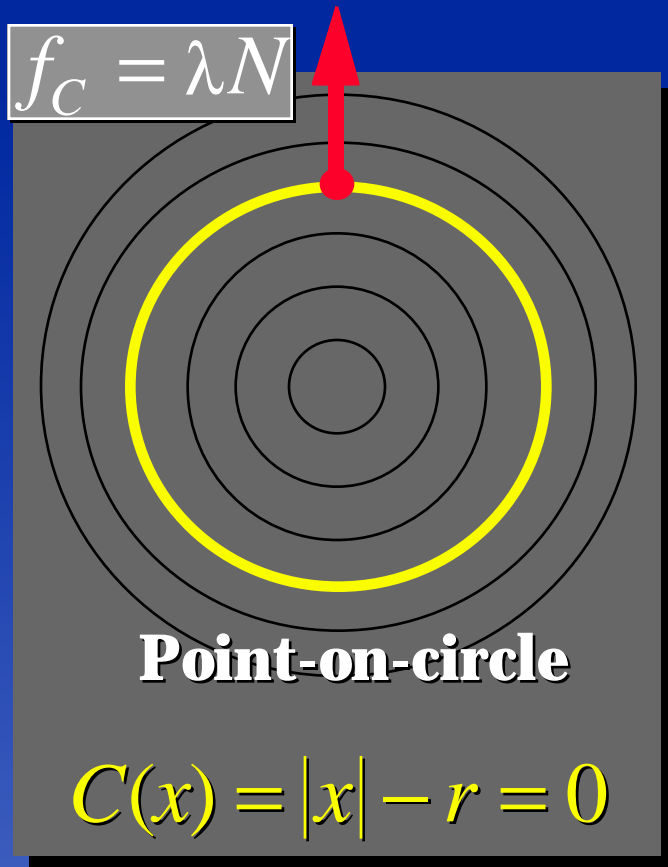
substitute

$$\dot{q} = W(f_G + J^T \lambda)$$

get

$$\dot{p} = JWf_G + JWJ^T \lambda$$

# Recalling Lagrange Multipliers



Then...

Assumption of *passive* constraints stipulated that  $f_C$  point along constraint gradient  $\frac{\partial C}{\partial x} = N$

Now...

Constraint gradients:  $\frac{\partial C}{\partial q} = J$

$f_C$  is a *linear combination* of constraint gradients:

$$f_C = J^T \lambda$$

and wind up with:

$$\dot{p} - JWf_G = JWJ^T\lambda$$

Diagram illustrating the matrix equation  $\dot{p} - JWf_G = JWJ^T\lambda$ . The equation is shown above a series of matrix blocks: a small grey rectangle representing  $\dot{p}$ , followed by an equals sign, a grey rectangle labeled  $J$ , a large grey rectangle labeled  $W$ , a grey rectangle labeled  $J^T$ , and a small grey rectangle labeled  $\lambda$ . Arrows point from the equation to the corresponding matrix blocks.

Can solve for  $\lambda$ , plug into formula for  $\dot{q}$

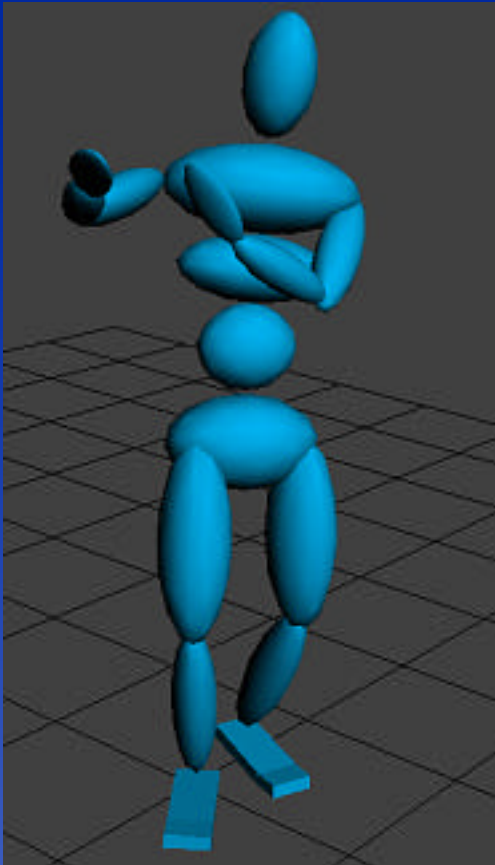
# Have we solved the scaling problem?

We get a velocity  $\dot{q}$  that obeys  
the “physical law”  $f = M\dot{v}$  ...

$M$  determines how the system responds  
to applied forces.

So it all depends on what  $M$  is.

# How well can we do?

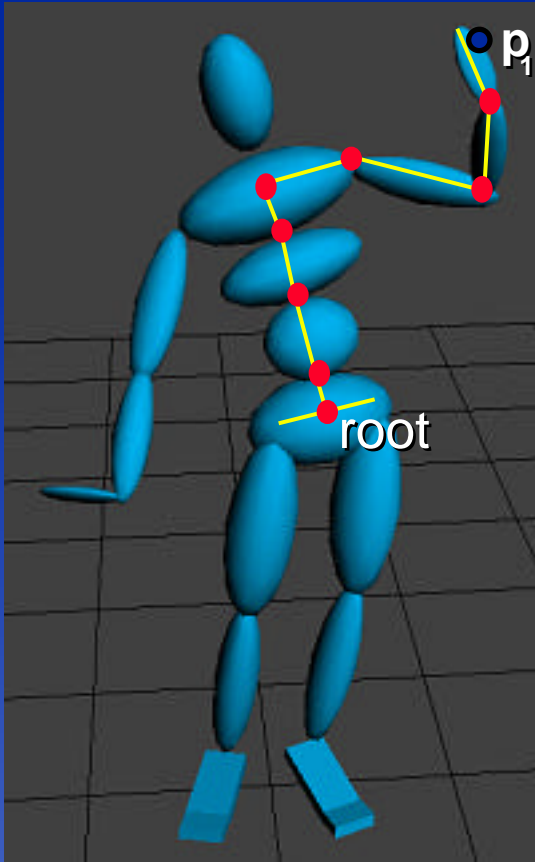


If approximate figure by rigid,  
uniform density, simple shapes...

Can derive and compute an  $M(q)$   
from minimizing kinetic energy  
of articulated figure

Minimum energy consumption is good!

# A word about $J$



$p_1$  only depends on ancestors in hierarchy

Therefore,  $J$  is sparse. Exact pattern depends on ordering of  $q$ .

Can compute all derivatives efficiently in recursive tree traversal



# What's in the online notes

- More detailed derivations
- Formula for computing Mass Matrix
- Pseudocode for tree traversal
- Bibliography

# That's Just the Beginning...

- One sided constraints → joint limits
- Hierarchical constraints
- Spring “muscles”
- Build complex constraint functions for higher level behavior