System Condition

Dr. Mitch Pryor

First, a simple example

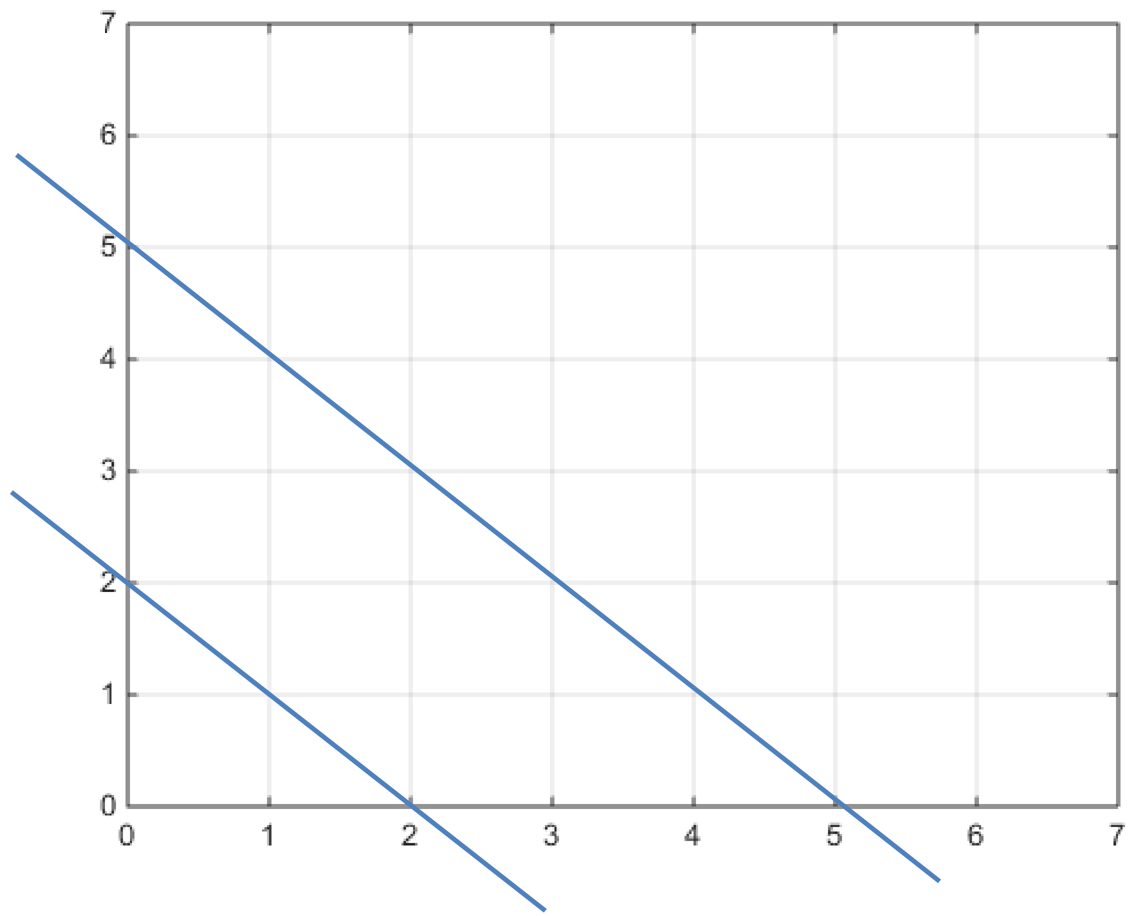
Start with two linear equations…

$$1.000x_1 + 1.000x_2 = 5.000$$
$$1.000x_1 + \beta x_2 = 2.000$$

First plot, assuming that β = 1.000

There is no solution **x** for these equations. Furthermore,

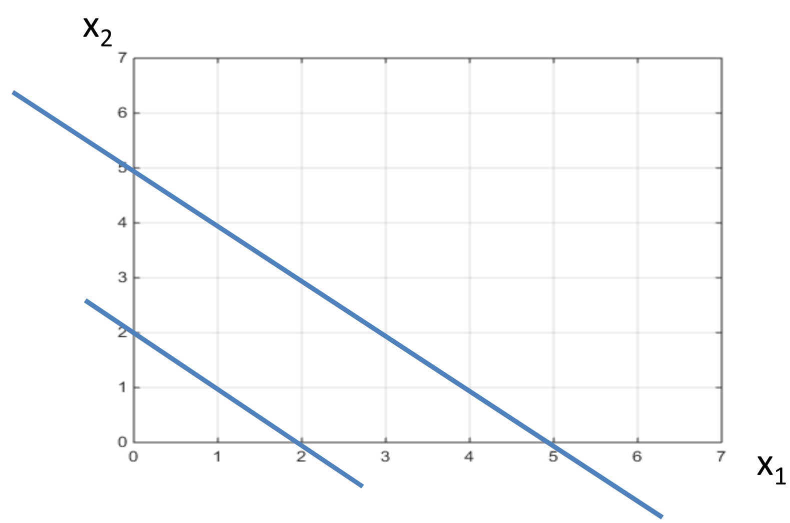
$$\det(\mathbf{A}) = \det\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = 0$$



Is Sol^n @ A or B if β = 0.999?

$$1.000x_1 + 1.000x_2 = 5.000$$
$$1.000x_1 + \beta x_2 = 2.000$$

Solution A



System **Condition**

ill conditioned system: small changes in system parameters cause "large" changes in the system output.
well conditioned system: small changes in system parameters produce "small" changes in the system output.

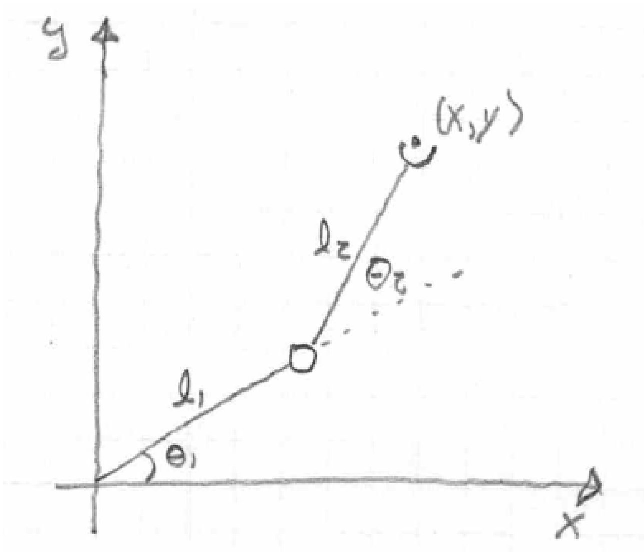
If a system (or system model) is ill or poorly conditioned, then, we may get very different results from small errors in our model or even from rounding errors in our calculations.

Can we quantify a system's condition so we determine if we trust our results?

Or if β = 1.001?

Solution B

Consider a 2DOF planar robot...



Two problems to solve:

Forward kinematics problem

Given: $\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ Find: $\mathbf{x}_{hand} = \begin{bmatrix} x \\ y \end{bmatrix}$

Solution: straight forward geometry.

$$x = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)$$

$$y = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2)$$

Inverse kinematics problem

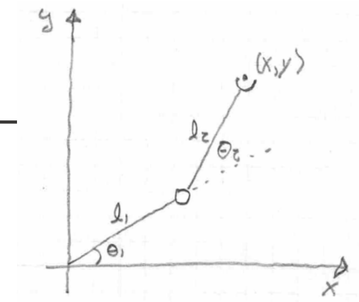
Given: $\mathbf{x}_{hand} = \begin{bmatrix} x \\ y \end{bmatrix}$ Find: $\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$

Solution: a little trickier…

- Can solve the equations above for joint angles
  - Multiple solutions
  - Method becomes untenable with higher number of joints.

Let's try something else…

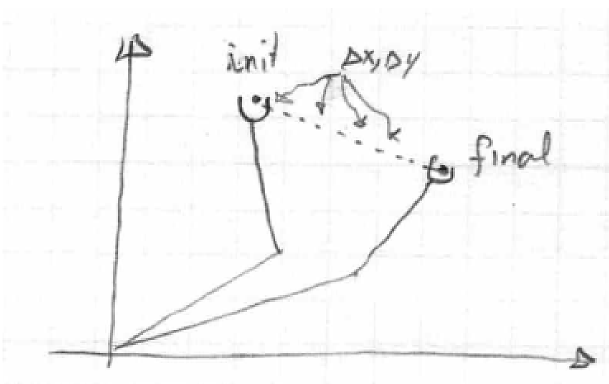
Consider a 2DOF planar robot…



Solution: start with our forward solution

$$x = f_x(\boldsymbol{\theta}) = l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2)$$
$$y = f_y(\boldsymbol{\theta}) = l_1\sin(\theta_1) + l_2\sin(\theta_1+\theta_2)$$

Instead of solving for joint angles at the final position, move the hand to the final position in incremental steps.



As Δ goes to zero, we are simply calculating the desired hand velocity.

To do this,, we again linearize the system.

$$\Delta x = \frac{\partial f_x}{\partial \theta_1}\Delta\theta_1 + \frac{\partial f_x}{\partial \theta_2}\Delta\theta_2$$

$$\Delta y = \frac{\partial f_y}{\partial \theta_1}\Delta\theta_1 + \frac{\partial f_y}{\partial \theta_2}\Delta\theta_2$$

Where…

$$\frac{\partial f_x}{\partial \theta_1} = -l_1\sin(\theta_1) - l_2\sin(\theta_1+\theta_2) \qquad \frac{\partial f_x}{\partial \theta_2} = -l_2\sin(\theta_1+\theta_2)$$

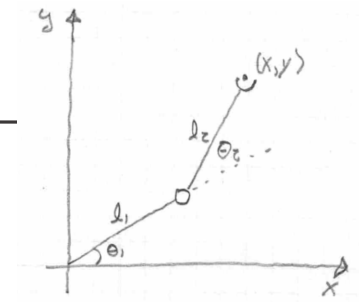
$$\frac{\partial f_y}{\partial \theta_1} = l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2) \qquad \frac{\partial f_y}{\partial \theta_2} = l_2\cos(\theta_1+\theta_2)$$

Which can be put in matrix form…

$$\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix} = \begin{bmatrix}\frac{\partial f_x}{\partial \theta_1} & \frac{\partial f_x}{\partial \theta_2}\\ \frac{\partial f_y}{\partial \theta_1} & \frac{\partial f_y}{\partial \theta_2}\end{bmatrix}\begin{bmatrix}\Delta\theta_1\\ \Delta\theta_2\end{bmatrix}$$

$$\dot{\mathbf{h}} = \underbrace{\mathbf{J}}_{\text{Jacobian Matrix}}\dot{\boldsymbol{\theta}}$$

Inverse Kinematics Problem

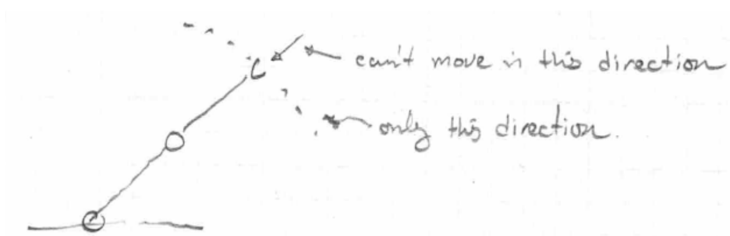


$$\dot{\mathbf{h}} = \mathbf{J}\dot{\boldsymbol{\theta}}$$

So to find the desired joint velocities, we only have to invert the Jacobian

$$\dot{\boldsymbol{\theta}} = \mathbf{J}^{-1}\dot{\mathbf{h}}$$

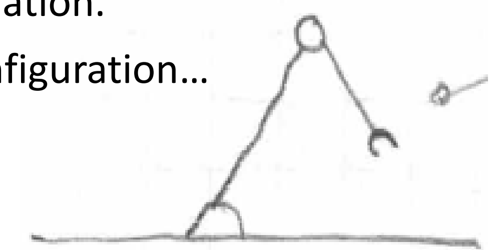
So we cannot solve this problem if the Jacobian is not invertible. For example, if the robot is in the following configuration.



But what if the robot is in this configuration?

In this case we can see that small motions in the *x* direction will require large joint motions. (i.e. the system is in an *ill*-conditioned configuration.

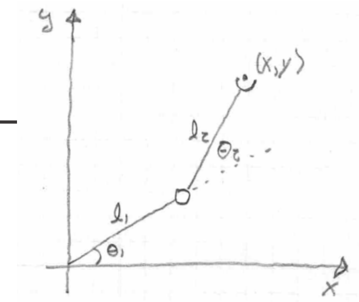
And if the robot is in this configuration…



Can generally move in any direction with reasonable joint velocities.

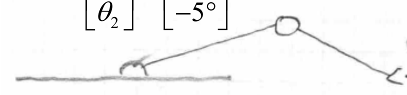
We can see this for the 2DOF planar robot fairly easily, but this visualization won't be possible for most systems AND it would be nice to have a quantifiable metric for system condition.

Inverse Kinematics Problem

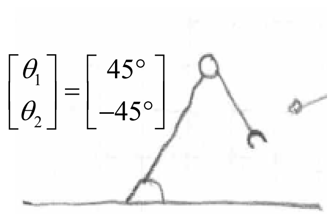


What about the determinant? Since 0 is bad, why not look at the value and see how close it is to zero. For example, if the length links are each 1 meter, then…

$$\begin{bmatrix}\theta_1\\ \theta_2\end{bmatrix}=\begin{bmatrix}5^\circ\\ -5^\circ\end{bmatrix}$$



$$\begin{bmatrix}\theta_1\\ \theta_2\end{bmatrix}=\begin{bmatrix}45^\circ\\ -45^\circ\end{bmatrix}$$



```
clear all;
len = 1; a1 = 45; a2 = -45;

J = [ -len*sind(a1)-len*sind(a1+a2) -len*sind(a1+a2);
len*cosd(a1)+len*cosd(a1+a2) len*cosd(a1+a2) ];

det(J)
```

```
ans =

   -0.0872
```

```
ans =

   -0.7071
```

So clearly the second configuration is farther from a singularity, but what if someone else uses feet instead of meters for the length?

```
ans =

   -0.8002
```

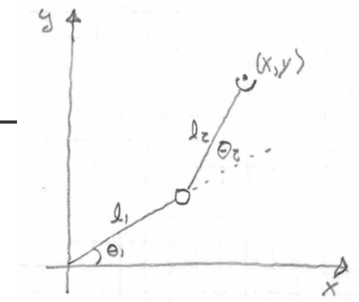
```
ans =

   -6.4919
```

The determinant value varies nonlinearly and is dependent on the units uses and may be further complicated if the equations mix units. (What if this was 3 DOF robot where the hand's orientation was also specified?)

Ideally we would like to find a unitless metric whose values are consistent for any system.

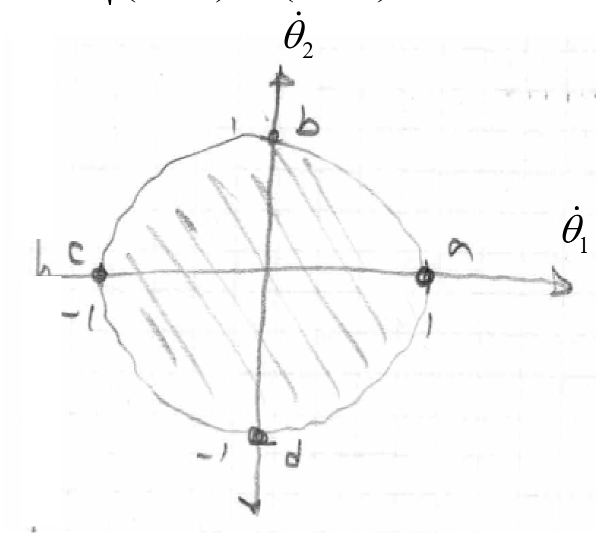
Let's define the condition number.

Inverse Kinematics Problem



Consider the set of all possible joint motions from a given initial joint configuration that meet the following requirement.

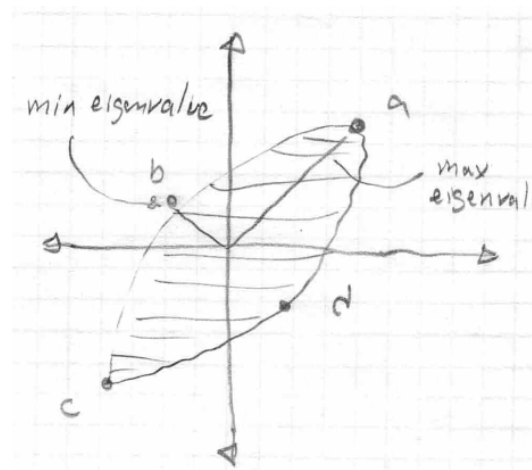
$$\sqrt{\left(\frac{\dot{\theta}_1}{\dot{\theta}_{1,\max}}\right)^2+\left(\frac{\dot{\theta}_2}{\dot{\theta}_{2,\max}}\right)^2}\leq 1$$



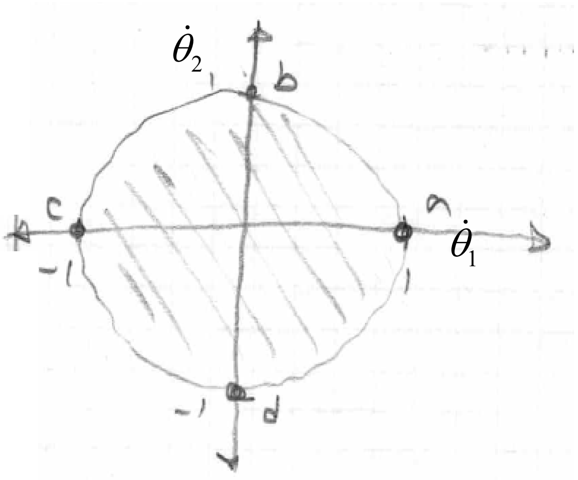
If we multiply every point in this set by the Jacobian, we get the set of all possible hand velocities is:

$$\mathbf{J}\dot{\boldsymbol{\theta}}=\dot{\mathbf{h}}$$

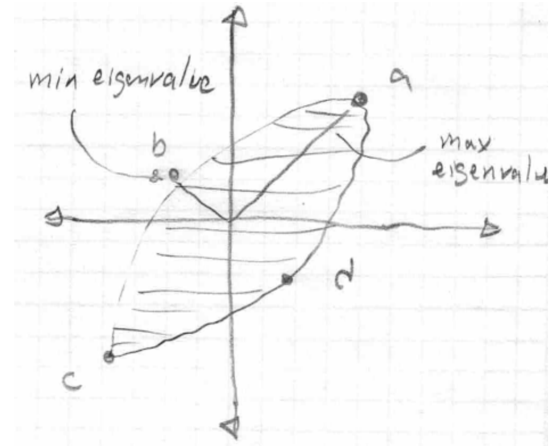
$$\Rightarrow \mathbf{J} \Rightarrow$$

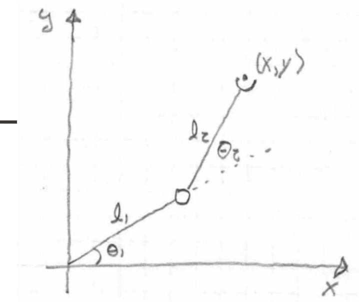


Condition Number



$$\Rightarrow \mathbf{J} \Rightarrow$$





Condition Number: $cond(\mathbf{J}) = c(\mathbf{J})$

$$= \frac{\lambda_{\max}}{\lambda_{\min}}$$

Worst possible condition number: ∞ (singular)

Best possible condition number: 1 (isotropic)

The condition metric can provide reasonable data on whether you can trust the results of your simulation or not.

General rule of thumb:

if cond(**J**) < 20 (or 200), system is well-conditioned

if cond(**J**) > 500 (or 5000), system is ill-conditioned