EEE 187 Robotics

LEGGED LOCOMOTION

- Legged robots are inspired by biological systems (animals, insects). Compared with wheeled locomotion, legged locomotion is more complex and more difficult to control. Legged locomotion needs to be stable.
  - Static stability: a statically stable robot can stand without falling over. The key is to have enough legs to provide support. Humans are not statically stable, standing up appears to be effortless, but we are actively using active control to achieve balance. For us, balancing is largely unconscious and must be learned.
  - Dynamic stability: dynamic stability allows a robot (or animal) to be stable while moving. A bicycle is a good example for dynamic stability. Another example is one legged hopping robots, they can hop in place or move to other destinations, and not fall over, but they cannot stop and stay standing without falling over.

EEE 187 Robotics

- Condition for stability
  - The condition for static stability is very simple: the robot's center of gravity must fall under the robot polygon of support. A comparison between static and dynamic walking is shown.
- Definitions
  - 1 Gait: manner of walking or running
  - 2 Gait: distinct sequence of lift and release events of the individual legs

Static walking Dynamic walking

EEE 187 Robotics

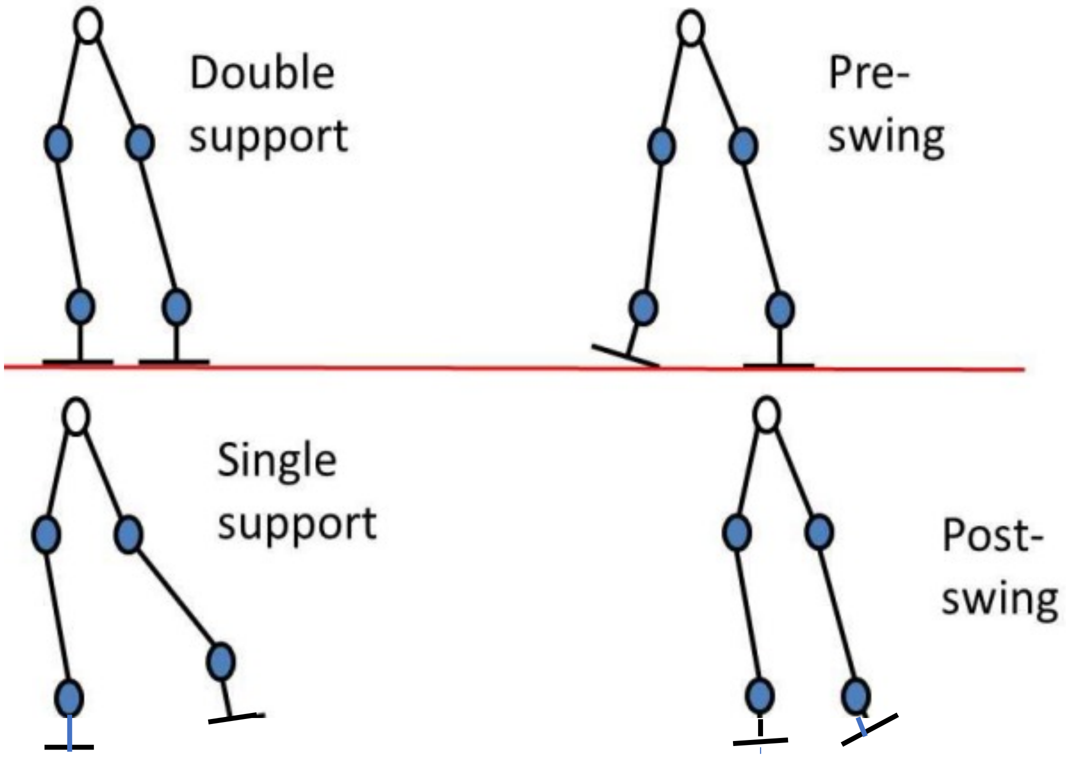
Walking is moving by putting forward each foot in turn, not having both feet off the ground at once. In running, there is a flight phase where the robot is not touching the ground.

- The swing leg is the leg performing the step.
- The stance foot is the foot that supports the weight of the robot.

Human walking is highly energy efficient. Walking is cyclic, it is a periodic process.

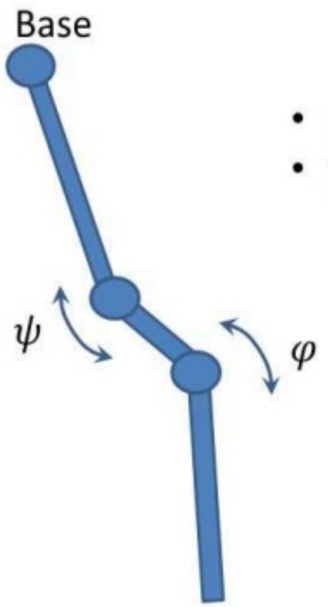
A walking gait consists of four different phases:

- Double support phase (DS): both feet are on the ground.
- Pre-swing phase: The heel of the rear foot is lifting. The biped is still in double support.
- Single support phase (SS): One foot only is on the ground
- Post-swing phase: The toe of the front foot is declining towards the floor; the biped is in double support because the heel of the front foot is touching the ground.



EEE 187 Robotics

- How many degrees of freedom are needed to move one leg?
  - At least two: one to lift the leg and the other one to move it forward.
- The gait depends on the number of legs.
  - What is the number of distinct events N for a walking robot machine with k legs? A distinct event sequence is a change from one stat to another and back



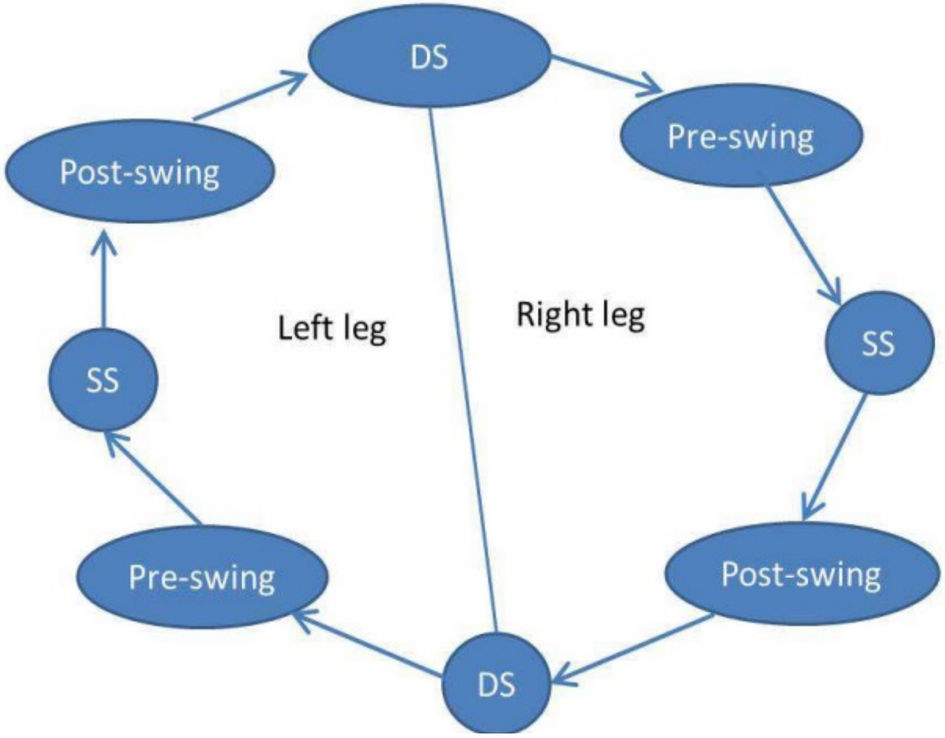
- φ: *Knee flexion angle*
- ψ: *Hip flexion angle*

Knee and hip flexion angles

EEE 187 Robotics

The single support phase is the key phase in the walking process. Most walking models focus on this phase. Biped robots usually have 12 DOF and three complex joints per leg.

Static walking is very slow with small steps, therefore we can ignore the dynamic forces. In static walking, the robots center of gravity is inside polygon of support.

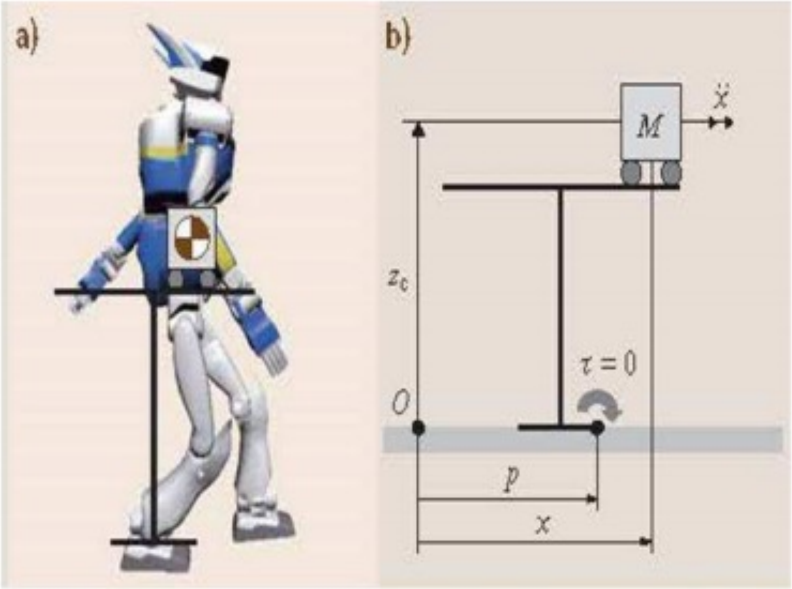


EEE 187 Robotics

Zero Moment Point and the cart–table model

Zero Moment Point (ZMP) is the point on the ground about which the net moment of the total forces applied to the biped is zero. The zero moment point can be used to prove the dynamic stability of the robot. The gait is stable if the ZMP remains within the foot-print polygons. The cart table model shown in figure 7 is used to model the single support phase where a) shows the walking robot and b) shows its simplified model. This model is used here to explain the ZMP. The model can be summarized as follows:

- The simplified model consists of a running cart on a massless table.
- The cart has mass M and its position is (x, z_c) corresponds to the center of mass of the robot.
- The table is assumed to have the same support polygon as the robot.



a) walking robot
b) its simplified model

Cart table model

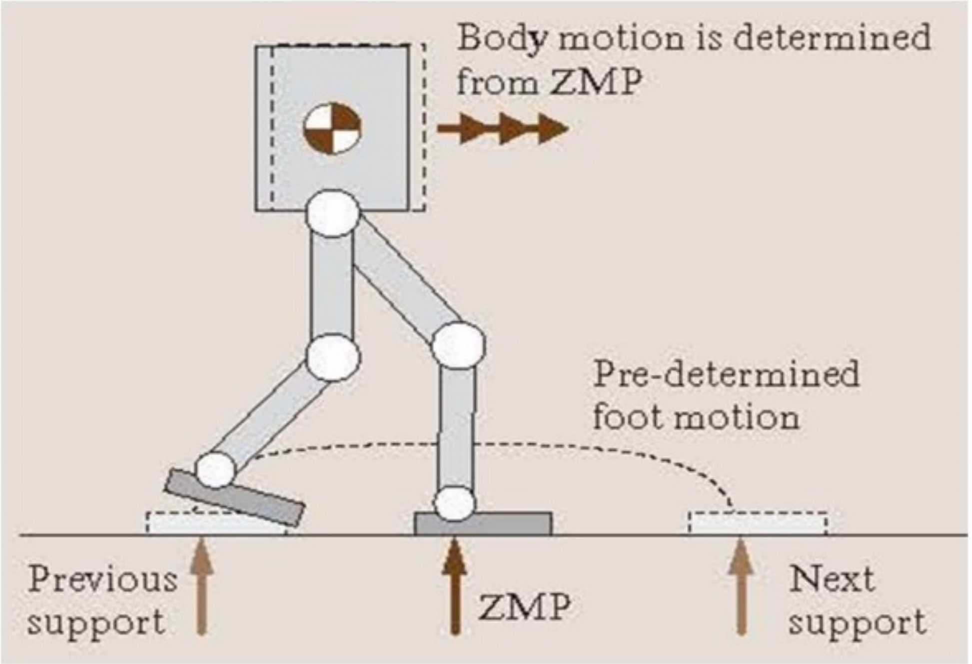
- Simplified model consists of a running cart on a massless table.
- The cart has mass *M* and its position is (*x*, *zc*) corresponds to the center of mass of the robot.
- The table is assumed to have the same support polygon as the robot.

EEE 187 Robotics

Algorithm

Generating the biped walking pattern using ZMP consists of three phases:

- Calculate the ZMP
- Generate trajectories for the feet such that locomotion is stable according to ZMP
- The motion of the joints is determined using the inverse kinematics from the body and the feet trajectories. Example: use of hip actuators.



Implementation of the ZMP scheme

EEE 187 Robotics

Performance criteria

- Duty factors: The duty factor is related to stability. Figure 17 shows the duty factor as function of the stability margin with the number of legs as the parameter.
  - Stability improves when the duty factor increases
  - Stability improves with the number of legs, but the most important "jump is between $N = 4$ and $N = 6$.

  Duty factors can be used to make the distinction between walks and runs. For example, for a biped, $\beta < 0.5$ for running.
- Froude number:

$$F_r = \frac{v^2}{gh}$$

  - v is the walking or running speed,
  - g is the acceleration due to gravity,
  - h is the height of hip joint from the ground. In particular, most animals change their gait from walking to running at a speed equivalent to a Froude number of $F_r = 1$.

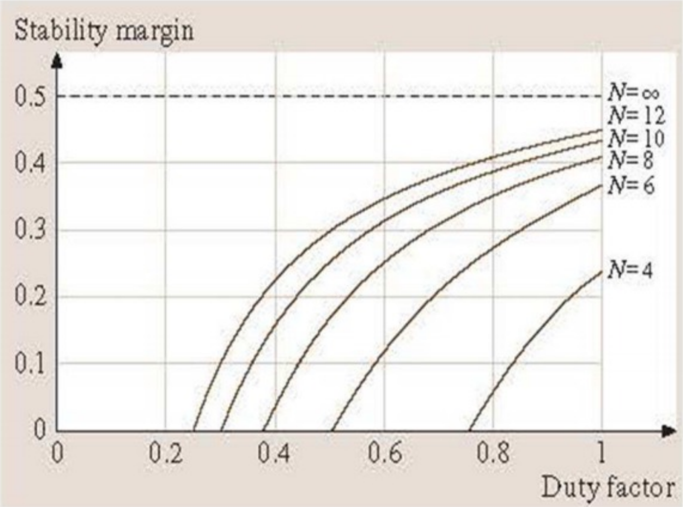


Fig. 17. Duty factor and stability margin

- Specific resistance: a dimensionless number that is used to evaluate the energy efficiency of a mobile robot

$$\varepsilon = \frac{E}{Mgd} \tag{11}$$

- E is the total energy consumption for a travel of distance d,
- M is the total mass of the vehicle,
- g is the acceleration due to gravity

The specific resistance is an indicator of the smoothness of the locomotion.

EEE 187 Robotics

Gait Diagram

The gait diagram is a way to describe the gait sequence of a multi-legged robot. The x-axis corresponds to the time and the y-axis corresponds to the leg number. Three important variables are defined:.

- Walk cycle time is T.
- Duty factor of leg i: β_i
- Phase of leg i: ϕ_i

The duty factor and the phase are defined below:

$$\beta_i = \frac{\text{support period of leg } i}{T}$$

and

$$\phi_i = \frac{\text{touchdown time of leg } i}{T}$$

Here's some fun links that demonstrate different robotic gaits

https://www.youtube.com/watch?v=R-PdPtqw78k

https://www.youtube.com/watch?v=MD48SLxwezg

https://www.youtube.com/watch?v=EuSR96U5NUY

https://www.youtube.com/watch?v=_dafK3E9WtE

https://www.youtube.com/watch?v=Dfke2byJknk

https://www.youtube.com/watch?v=VvkIdCKlL54

https://www.youtube.com/watch?v=OBGHU-e1kc0&feature=youtu.be

Voltage → x
Distance → y

x	-3	-2	-1	-0.2	1	3
y	0.9	0.8	0.4	0.2	0.1	0

$$y = a_0 + a_1 x + \ldots + a_k x^k,$$

the residual is given by

$$R^2 \equiv \sum_{i=1}^{n} [y_i - (a_0 + a_1 x_i + \ldots + a_k x_i^k)]^2.$$

residual = R^2 =difference between the vendor supplied distance information and the value calculated from our polynomial

$$R^2 = [0.9 - (a_0 + a_1 (-3) + a_2 (-3)^2)]^2 + [0.8 - (a_0 + a_1 (-2) + a_2 (-2)^2)]^2 + [0.4 - (a_0 + a_1 (-1) + a_2 (-1)^2)]^2$$
$$+ [0.2 - (a_0 + a_1 (-0.2) + a_2 (-0.2)^2)]^2 + [0.1 - (a_0 + a_1 (1) + a_2 (1)^2)]^2 + [0 - (a_0 + a_1 (3) + a_2 (3)^2)]^2$$

$d(R^2)/d(a_0) = 0$
$d(R^2)/d(a_1) = 0$
$d(R^2)/d(a_2) = 0$

EEE 187 Robotics

Sensor Calibration using Least Squares Fit

Robots typically make use of sensors that generate voltage levels that correlate to measurable parameters within the robot's environment. A method known as the Least Squares Fit is typically used to find the polynomial coefficients (a0….an) that can be used within a standard polynomial to approximate the relationship between a sensor's generated voltage and the measurable parameter that correlates to that voltage (for example, a light source of obstacle's distance from the sensor)

In the equation below x = voltage and y= distance. The sensor vendor has given us a table that describes voltages that correspond to various distances. We (now) need to calculate the coefficients (a_0, a_1, a_2,….a_n) that can be used to approximately represent the distance vs voltage relationship that has been described by the vendor.

$$y = a_0 + a_1 x + \ldots + a_k x^k,$$

the residual is given by

$$R^2 \equiv \sum_{i=1}^{n} \left[y_i - \left(a_0 + a_1 x_i + \ldots + a_k x_i^k \right) \right]^2.$$

residual = R^2 =difference between the vendor supplied distance information and the value calculated from our polynomial

$$\frac{\partial(R^2)}{\partial a_0} = -2 \sum_{i=1}^{n} \left[y - \left(a_0 + a_1 x + \ldots + a_k x^k \right) \right] = 0$$

$$\frac{\partial(R^2)}{\partial a_1} = -2 \sum_{i=1}^{n} \left[y - \left(a_0 + a_1 x + \ldots + a_k x^k \right) \right] x = 0$$

$$\frac{\partial(R^2)}{\partial a_k} = -2 \sum_{i=1}^{n} \left[y - \left(a_0 + a_1 x + \ldots + a_k x^k \right) \right] x^k = 0.$$

We now need find values for a_i that will result in $\frac{\partial(R^2)}{\partial a_1} = 0$ as describe above

EEE 187 Robotics

Sensor Calibration using Least Squares Fit

Minimizing the residuals requires that the polynomial coefficients (ai….an) be described by the relationships;

$$a_0\, n + a_1 \sum_{i=1}^{n} x_i + \ldots + a_k \sum_{i=1}^{n} x_i^k = \sum_{i=1}^{n} y_i$$

$$a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 + \ldots + a_k \sum_{i=1}^{n} x_i^{k+1} = \sum_{i=1}^{n} x_i\, y_i$$

$$a_0 \sum_{i=1}^{n} x_i^k + a_1 \sum_{i=1}^{n} x_i^{k+1} + \ldots + a_k \sum_{i=1}^{n} x_i^{2k} = \sum_{i=1}^{n} x_i^k\, y_i$$

Add up all of the y values of all of the data points

Sum up the products of x and y for all data points

or, in matrix form

n= total number of data points

Add up all of the x values of all of the data points

Square the x values of all of the data points and then add them together

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & \cdots & \sum_{i=1}^{n} x_i^k \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \cdots & \sum_{i=1}^{n} x_i^{k+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_i^k & \sum_{i=1}^{n} x_i^{k+1} & \cdots & \sum_{i=1}^{n} x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i\, y_i \\ \vdots \\ \sum_{i=1}^{n} x_i^k\, y_i \end{bmatrix}.$$

EEE 187 Robotics

Sensor Calibration using Least Squares Fit

The following example demonstrates how to develop a 2nd order polynomial curve fit for the following dataset:

Voltage

Distance

x	-3	-2	-1	-0.2	1	3
y	0.9	0.8	0.4	0.2	0.1	0

This dataset has $N = 6$points and for a 2nd order polynomial $k = 2$. As shown in the previous section, application of the least of squares method provides the following linear system.

(-3-2-1-0.2+1+3 =-2.2)

$(-3)^2 + (-2)^2 + (-1)^2 + (-0.2)^2 + (1)^2 + (3)^2 = 24.02$

0.9 + 0.8 + 0.4 + 0.2 +0.1 +0 =2.4

$$\begin{bmatrix} 6 & -2.2 & 24.04 \\ -2.2 & 24.04 & -8.008 \\ 24.04 & -8.008 & 180.0016 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 2.4 \\ -4.64 \\ 11.808 \end{bmatrix}$$

$$\begin{bmatrix} 6 & -2.2 & 24.04 \\ -2.2 & 24.04 & -8.008 \\ 24.04 & -8.008 & 180.0016 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 2.4 \\ -4.64 \\ 11.808 \end{bmatrix}$$

EEE 187 Robotics

Sensor Calibration using Least Squares Fit

Using Cramer's rule to solve the system we generate each of the matrices M_i by taking the matrix M and substituting the column vector b into the i$^{\text{th}}$ column, for example M_0 and M_1 would be:

$$M_0 = \begin{bmatrix} 2.4 & -2.2 & 24.04 \\ -4.64 & 24.04 & -8.008 \\ 11.808 & -8.008 & 180.0016 \end{bmatrix}$$

$$M_1 = \begin{bmatrix} 6 & 2.4 & 24.04 \\ -2.2 & -4.64 & -8.008 \\ 24.04 & 11.808 & 180.0016 \end{bmatrix}$$

EEE 187
Robotics

Sensor Calibration using Least Squares Fit

Once these matrices have been formed the determinant for each of the square matrices $M, M_0, M_1 \text{and} M_2$ can be calculated and utilised to determine the polynomial coefficients as follows:

$$a_0 = \frac{det(M_0)}{det(M)} = \frac{2671.20}{11661.27} = 0.2291$$
$$a_1 = \frac{det(M_1)}{det(M)} = \frac{-1898.46}{11661.27} = -0.1628$$
$$a_2 = \frac{det(M_2)}{det(M)} = \frac{323.76}{11661.27} = 0.0278$$

The polynomial regression of the dataset may now be formulated using these coefficients.

$$y = 0.0278x^2 - 0.1628x + 0.2291$$

EEE 187
Robotics

Sensor Calibration using Least Squares Fit

n= total number of data points

Add up all of the x values of all of the data points

Square the x values of all of the data points and then add them together

or, in matrix form

Sum up the products of x and y for all data points

points

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & \cdots & \sum_{i=1}^{n} x_i^k \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \cdots & \sum_{i=1}^{n} x_i^{k+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_i^k & \sum_{i=1}^{n} x_i^{k+1} & \cdots & \sum_{i=1}^{n} x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \\ \vdots \\ \sum_{i=1}^{n} x_i^k y_i \end{bmatrix}.$$

$Y = 2 + X + X^2$

X,Y
1,4
2,8
3,14
4,22
5,32

X^4	X^3	X^2	X
1	1	1	1
16	8	4	2
81	27	9	3
256	64	16	4
625	125	25	5

XY	X^2Y
4	4
16	32
42	126
88	352
160	800

EEE 187

Robotics

Sensor Calibration using Least Squares Fit

$$\begin{pmatrix} 5 & 15 & 55 \\ 15 & 55 & 225 \\ 55 & 225 & 979 \end{pmatrix} \begin{pmatrix} a0 \\ a1 \\ a2 \end{pmatrix} = \begin{pmatrix} 80 \\ 310 \\ 1314 \end{pmatrix}$$