## How to Simulate a Trebuchet Part 3: The Floating-Arm Trebuchet

The illustration above gives a diagram of a "floating-arm" trebuchet. The floating-arm type is distinct from the ordinary trebuchet in that its arm has no fixed pivot; that is, it "floats" during a launch. The main reason for doing this is to permit the counterweight to move in a straight line downwards, rather than revolving about a fixed point. In doing this, a higher proportion of the gravitational potential energy of the counterweight is available for increasing the kinetic energy of the projectile, rather than increasing the rotational kinetic energy of the counterweight.

The figure above right shows a bench-scale floating-arm trebuchet, courtesy of Woodworkers Workshop. There are two rollers which are important for the motion of the arm. The fixed roller is attached to the end of the horizontal rail, and the arm roller is attached to the arm near the counterweight.
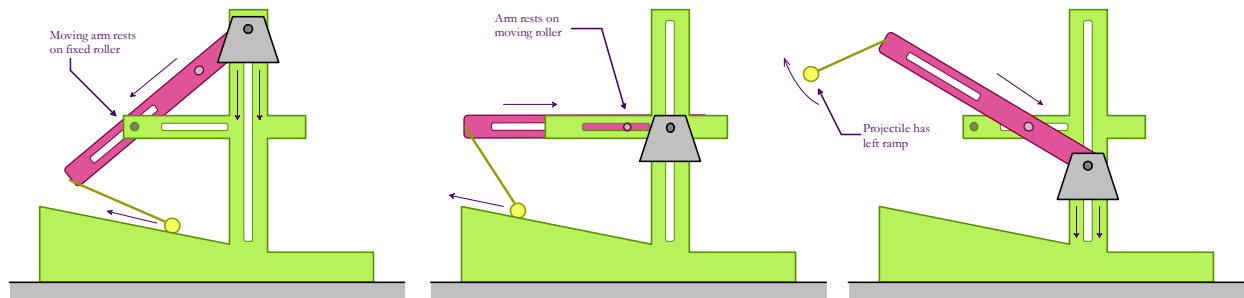
The Motion of the Floating-Arm Trebuchet

The launch sequence of the floating-arm trebuchet is shown in the three figures above. First, the throwing arm rests on a fixed roller, which is pinned to the horizontal support. As the counterweight descends, the arm slides over the fixed roller. When the throwing arm is roughly horizontal, the moving roller contacts the horizontal support. The combination of the moving roller support and the vertical counterweight slot creates a "whipping" effect that significantly increases the angular velocity of the throwing arm. In the final stage of the sequence, the projectile leaves the launch ramp and swings around the end of the throwing arm. The sling release mechanism works in the same way as a traditional trebuchet, and the projectile is released when the sling ropes become parallel with the hook.

To summarize, we have three distinct situations to consider when modeling the floating-arm trebuchet: arm on fixed roller, arm on moving roller and projectile on ramp. Each of these will be dealt with using constraint equations, as shown below.

Modeling the Floating-Arm Trebuchet

There are a number of different models we could choose for the floating-arm trebuchet. When the projectile is on the ramp, the system has one degree of freedom (DOF). For example, specifying the angle of the throwing arm constrains the configuration of the rest of the system. When the projectile is above the ramp, the system has two degrees of freedom (e.g. throwing arm angle and sling angle). If we choose, we can develop the equations of motion using only these one or two generalized coordinates – this is the 2DOF model. We may also choose to retain the full set of five degrees of freedom (height of counterweight, x and y coordinates of projectile, arm angle and sling angle) and use constraint equations to eliminate the redundant coordinates.

Intuitively, one might expect the 2DOF approach to lead to a much simpler numerical implementation, with much more compact code and faster execution. Unfortunately, the 2DOF model seems to encounter difficulties in switching from one set of constraints to another, especially when the moving roller begins to rest on the fixed horizontal track. The resulting animations tend to be rather jerky, or sometimes unstable. This is the reason for adopting the full 5DOF model with constraints that is developed below. The reasons behind the poor performance of the 2DOF model are unclear to this author, and are a subject for further research.
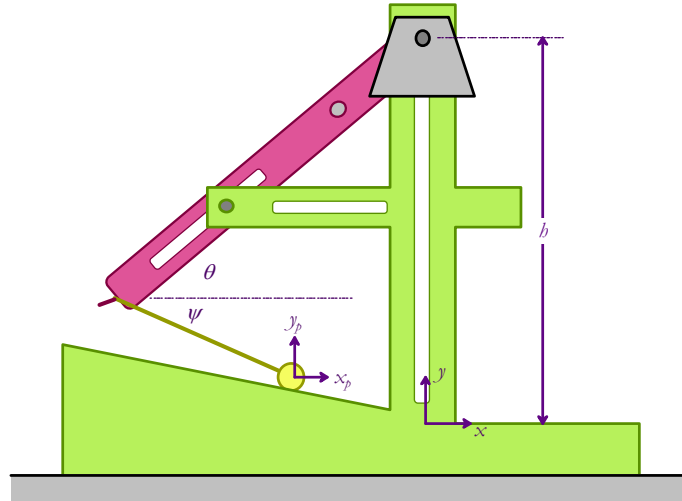
The generalized coordinates are shown above. To keep the model simple we will neglect the mass of the throwing arm. This leaves only two masses: the counterweight and the projectile. The simplest place to locate the global origin is at the intersection of the ramp and the vertical track. The location of the counterweight is given by $(0, h)$ and the projectile is (x_p, y_p). The throwing arm makes an angle θ with the horizontal and the sling is at angle ψ (positive angles are counterclockwise).

Since the counterweight is constrained to ride in the vertical track, its x coordinate is always zero. Eliminating it from the set of generalized coordinates leaves us with a total of five. When the projectile is on the ramp, we need four constraint equations, and when it is above the ramp we need three.

Equations of Motion for the Floating-Arm Trebuchet
The simplest way to develop equations of motion for this system is to use Lagrange's equations, with constraints enforced using Lagrange multipliers. The first step is to create position vectors to each body in the system.

$$\mathbf{r}_1 = \begin{Bmatrix} 0 \\ h \end{Bmatrix} \qquad\qquad \mathbf{r}_2 = \begin{Bmatrix} x_p \\ y_p \end{Bmatrix}$$

And the velocity of each mass is found by differentiating with respect to time.

$$\mathbf{v}_1 = \begin{Bmatrix} 0 \\ \dot{h} \end{Bmatrix} \qquad\qquad \mathbf{v}_2 = \begin{Bmatrix} \dot{x}_p \\ \dot{y}_p \end{Bmatrix}$$

Thus, the total kinetic energy of the system is

$$T = \frac{1}{2}m_1\dot{h}^2 + \frac{1}{2}I\dot{\theta}^2 + \frac{1}{2}m_2\left(\dot{x}_p^2 + \dot{y}_p^2\right)$$

where I is the moment of inertia of the throwing arm. The potential energy of the system is simple to formulate with our chosen set of generalized coordinates

$$V = m_1 g h + m_2 g y_p$$

Thus, the total Lagrangian of the system is

$$\mathcal{L} = T - V$$
$$= \frac{1}{2} m_1 \dot{h}^2 + \frac{1}{2} I \dot{\theta}^2 + \frac{1}{2} m_2 (\dot{x}_p^2 + \dot{y}_p^2) - m_1 g h - m_2 g y_p$$

Unconstrained Lagrange's Equations

To find the unconstrained Lagrange's equations, we must take partial derivatives of the Lagrangian with respect to each generalized coordinate.

$$\frac{\partial \mathcal{L}}{\partial \dot{h}} = m_1 \dot{h} \qquad\qquad \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{h}}\right) = m_1 \ddot{h} \qquad\qquad \frac{\partial \mathcal{L}}{\partial h} = -m_1 g$$

$$\frac{\partial \mathcal{L}}{\partial \dot{\theta}} = I \dot{\theta} \qquad\qquad \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}\right) = I \ddot{\theta} \qquad\qquad \frac{\partial \mathcal{L}}{\partial \theta} = 0$$

$$\frac{\partial \mathcal{L}}{\partial \dot{x}_p} = m_2 \dot{x}_p \qquad\qquad \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}_p}\right) = m_2 \ddot{x}_p \qquad\qquad \frac{\partial \mathcal{L}}{\partial x_p} = 0$$

$$\frac{\partial \mathcal{L}}{\partial \dot{y}_p} = m_2 \dot{y}_p \qquad\qquad \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{y}_p}\right) = m_2 \ddot{y}_p \qquad\qquad \frac{\partial \mathcal{L}}{\partial y_p} = -m_2 g$$

$$\frac{\partial \mathcal{L}}{\partial \dot{\psi}} = 0 \qquad\qquad \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\psi}}\right) = 0 \qquad\qquad \frac{\partial \mathcal{L}}{\partial \psi} = 0$$

Using these derivatives, we may solve for the inertia matrix and applied force vectors on the system (see the preceding document for further details).

$$\mathbf{M} = \begin{bmatrix} m_1 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & m_2 & 0 & 0 \\ 0 & 0 & 0 & m_2 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad\qquad \mathbf{g} + \mathbf{h} = \begin{Bmatrix} -m_1 g \\ 0 \\ 0 \\ -m_2 g \\ 0 \end{Bmatrix} \qquad (1)$$

Constraint Equations

There are four constraint conditions on the system, each of which will be explained in detail below.

1. The arm rests on the fixed roller at the beginning of the launch sequence.
2. In the middle of the launch sequence, the moving roller (attached to the arm) comes to rest on the horizontal track.
3. At the beginning of the launch sequence, and for a short period of time, the projectile slides along the launch ramp.
4. The end of the throwing arm is connected to the top end of the sling.
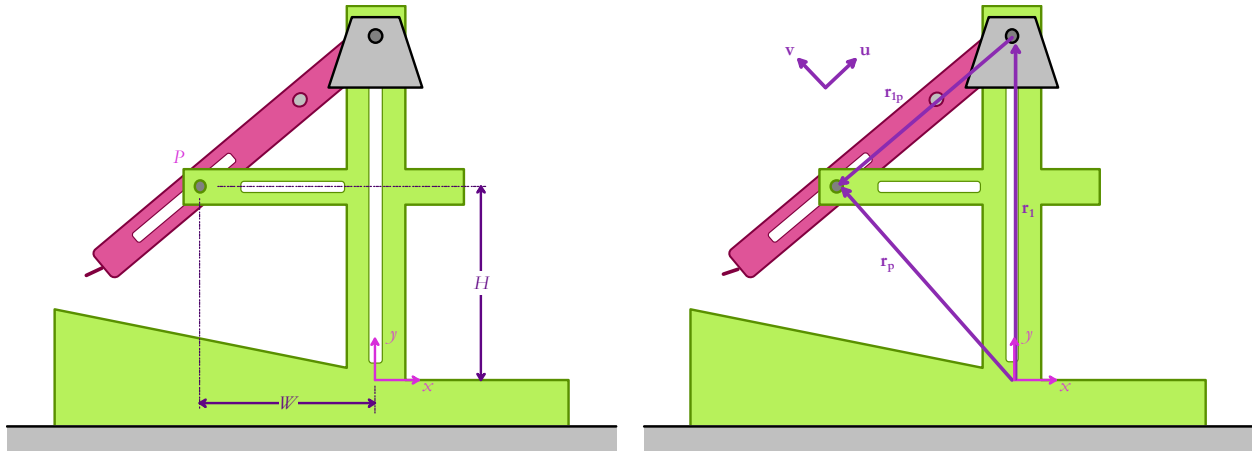
1. Arm rests on fixed roller

The figure above shows the throwing arm poised for launch. To simplify matters, we choose to model the constraint as a fixed pin in a moving slot. In reality, the arm is offset vertically by the radius of the roller and the thickness of the arm, but the pin-in-slot serves as a good first approximation. The fixed pin is at a vertical distance H and horizontal distance W from the origin. Define a vector $\mathbf{u}$ parallel to the arm, and $\mathbf{v}$ perpendicular to the arm. Both of these vectors rotate with the arm, and $\mathbf{v}$ can be written as

$$\mathbf{v} = \left\{ \begin{matrix} -\sin\theta \\ \cos\theta \end{matrix} \right\}$$

In order for the pin to remain in the slot, the vector from the counterweight to the fixed pin must be perpendicular to $\mathbf{v}$:
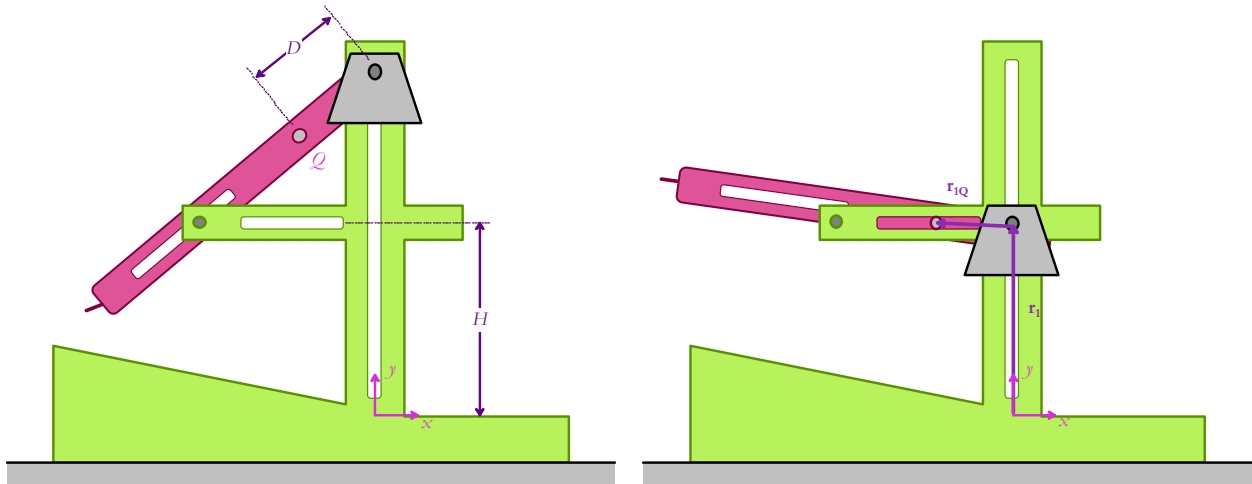
$$\mathbf{r}_{1P} \cdot \mathbf{v} = 0$$

where $\mathbf{r}_{1P}$ is found by subtracting $\mathbf{r}_P$ from $\mathbf{r}_1$.

$$\mathbf{r}_{1P} = \left\{ \begin{matrix} -W \\ H - h \end{matrix} \right\}$$

Thus, the fixed pin in moving slot can be written

$$W \sin\theta + (H - h)\cos\theta = 0$$

2. Arm rests on moving roller

Approximately midway through the launch sequence the arm roller comes to rest on the horizontal track. This is also modeled as a pin in slot, though here the slot is fixed and the pin (at point Q) moves with the arm. In order for the pin to remain in the slot, the vertical coordinate of point Q (on the arm) must remain at the height of the slot, H. This can be written

$$h - D \sin \theta - H = 0$$
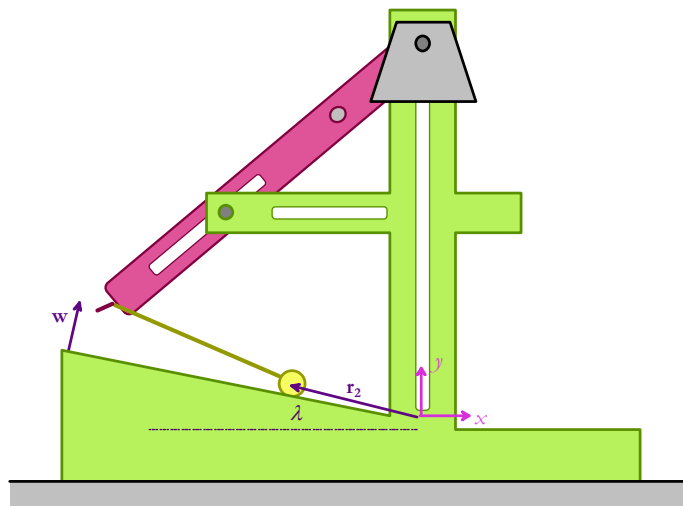
3. Projectile Slides on Ramp

The figure above shows the trebuchet at the beginning of the launch cycle, with the projectile resting on the ramp. As before, we model this as a pin (the projectile) moving in a fixed slot (the ramp). Define a vector $\mathbf{w}$ perpendicular to the ramp, whose angle is λ. Since we have defined the origin as the intersection of the ramp and the vertical slot, the constraint equation is particularly simple to write
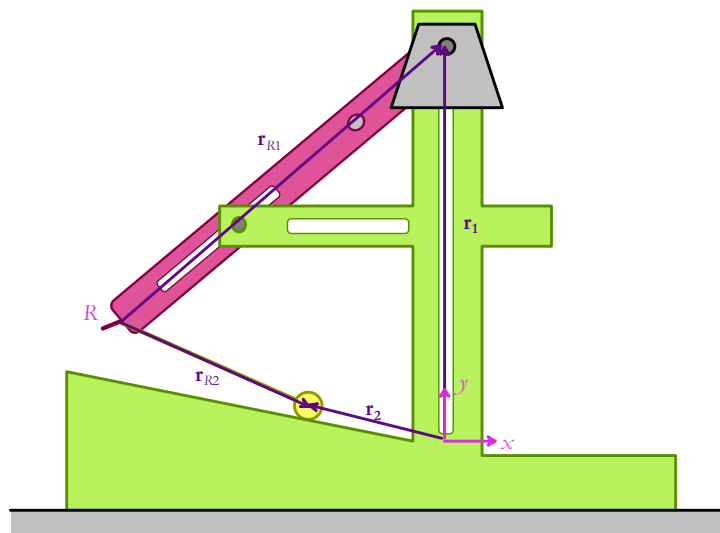
$$\mathbf{r}_2 \cdot \mathbf{w} = 0$$

where

$$\mathbf{w} = \begin{Bmatrix} \sin \lambda \\ \cos \lambda \end{Bmatrix}$$

Completing the dot product gives

$$x_p \sin \lambda + y_p \cos \lambda = 0$$

Note that if there is no ramp (i.e., if the ramp is horizontal), the above equation simplifies to

$$y_p = 0$$

4. Pin joint between arm and sling

The figure above gives a vector diagram of the constraint between the arm and the sling. Adding the vectors in the loop gives

$$\mathbf{r}_1 - \mathbf{r}_{R1} + \mathbf{r}_{R2} - \mathbf{r}_2 = 0$$

This can be written longhand as

$$\begin{Bmatrix} -L_2 \cos \theta - x_p + L_3 \cos \psi \\ h - L_2 \sin \theta - y_p + L_3 \sin \psi \end{Bmatrix} = 0$$

where L_2 is the length of the arm and L_3 is the length of the sling.

A "Menu" of Constraint Equations

Only a subset of the constraint equations are active at any time during the launch, although the constraint between sling and arm is always present. There are four possible constraint configurations for the trebuchet:

1. Arm-sling constraint, arm on fixed roller, projectile on ramp.
2. Arm-sling constraint, arm roller on fixed track, projectile on ramp.
3. Arm-sling constraint, arm on fixed roller, projectile above ramp.
4. Arm-sling constraint, arm roller on fixed track, projectile above ramp.

Determining which configuration the trebuchet is in at a given moment is one of the interesting aspects of modeling the system. As stated earlier, the arm-sling constraint is always active. The arm switches from being supported by the fixed roller to the moving roller when the height of the moving pivot, Q, is less than or equal to the height of the horizontal track. In other words:

$$h - D \sin \theta > H \rightarrow \text{arm on fixed roller}$$
$$h - D \sin \theta \leq H \rightarrow \text{arm roller on fixed track}$$

As with the fixed-arm trebuchet, the projectile is said to have left the ramp when the Lagrange multiplier corresponding to its constraint changes sign. We now have enough information to simulate the floating arm trebuchet by solving the equations of motion given in Equation (42) in the previous document:

$$\begin{bmatrix} \mathbf{M} & \dfrac{\partial \mathbf{f}^T}{\partial \mathbf{q}} \\ \dfrac{\partial \mathbf{f}}{\partial \mathbf{q}} & 0 \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{Bmatrix} = \begin{Bmatrix} \mathbf{g} + \mathbf{h} \\ \boldsymbol{\gamma} \end{Bmatrix}$$

The tables on the following page give the constraint vectors, $\mathbf{f}$, the Jacobian matrices, $\dfrac{\partial \mathbf{f}}{\partial \mathbf{q}}$, and the acceleration vectors, $\boldsymbol{\gamma}$, for each of the configurations listed above. The mass matrix and force vector are found in Equation (1) above; these do not depend upon the current constraint situation. At each time step, the code solves for accelerations, $\ddot{\mathbf{q}}$, and constraint forces $\boldsymbol{\lambda}$. It then checks to ensure that the constraint vector, $\mathbf{f}$, is satisfied. If it is not, a simple Newton-Raphson routine is used to bring the constraints into compliance. This process continues until a launch condition has been reached.

Constraint Equations, Jacobian Matrices and Acceleration Vectors

	Constraint Vector	Acceleration Vector
1	$\begin{Bmatrix} -L_2\cos\theta - x_p + L_3\cos\psi \\ h - L_2\sin\theta - y_p + L_3\sin\psi \\ W\sin\theta + (H-h)\cos\theta \\ x_p\sin\lambda + y_p\cos\lambda \end{Bmatrix}$	$\begin{Bmatrix} -L_2\dot\theta^2\cos\theta + L_3\dot\psi^2\cos\psi \\ -L_2\dot\theta^2\sin\theta + L_3\dot\psi^2\sin\psi \\ -2\dot h\dot\theta\sin\theta + \dot\theta^2[W\sin\theta + (H-h)\cos\theta] \\ 0 \end{Bmatrix}$
2	$\begin{Bmatrix} -L_2\cos\theta - x_p + L_3\cos\psi \\ h - L_2\sin\theta - y_p + L_3\sin\psi \\ h - D\sin\theta - H \\ x_p\sin\lambda + y_p\cos\lambda \end{Bmatrix}$	$\begin{Bmatrix} -L_2\dot\theta^2\cos\theta + L_3\dot\psi^2\cos\psi \\ -L_2\dot\theta^2\sin\theta + L_3\dot\psi^2\sin\psi \\ -D\dot\theta^2\sin\theta \\ 0 \end{Bmatrix}$
3	$\begin{Bmatrix} -L_2\cos\theta - x_p + L_3\cos\psi \\ h - L_2\sin\theta - y_p + L_3\sin\psi \\ W\sin\theta + (H-h)\cos\theta \end{Bmatrix}$	$\begin{Bmatrix} -L_2\dot\theta^2\cos\theta + L_3\dot\psi^2\cos\psi \\ -L_2\dot\theta^2\sin\theta + L_3\dot\psi^2\sin\psi \\ -2\dot h\dot\theta\sin\theta + \dot\theta^2[W\sin\theta + (H-h)\cos\theta] \end{Bmatrix}$
4	$\begin{Bmatrix} -L_2\cos\theta - x_p + L_3\cos\psi \\ h - L_2\sin\theta - y_p + L_3\sin\psi \\ h - D\sin\theta - H \end{Bmatrix}$	$\begin{Bmatrix} -L_2\dot\theta^2\cos\theta + L_3\dot\psi^2\cos\psi \\ -L_2\dot\theta^2\sin\theta + L_3\dot\psi^2\sin\psi \\ -D\dot\theta^2\sin\theta \end{Bmatrix}$

	Jacobian Matrix
1	$\begin{bmatrix} 0 & L_2\sin\theta & -1 & 0 & -L_3\sin\psi \\ 1 & -L_2\cos\theta & 0 & -1 & L_3\cos\psi \\ -\cos\theta & W\cos\theta - (H-h)\sin\theta & 0 & 0 & 0 \\ 0 & 0 & \sin\lambda & \cos\lambda & 0 \end{bmatrix}$
2	$\begin{bmatrix} 0 & L_2\sin\theta & -1 & 0 & -L_3\sin\psi \\ 1 & -L_2\cos\theta & 0 & -1 & L_3\cos\psi \\ 1 & -D\cos\theta & 0 & 0 & 0 \\ 0 & 0 & \sin\lambda & \cos\lambda & 0 \end{bmatrix}$
3	$\begin{bmatrix} 0 & L_2\sin\theta & -1 & 0 & -L_3\sin\psi \\ 1 & -L_2\cos\theta & 0 & -1 & L_3\cos\psi \\ -\cos\theta & W\cos\theta - (H-h)\sin\theta & 0 & 0 & 0 \end{bmatrix}$
4	$\begin{bmatrix} 0 & L_2\sin\theta & -1 & 0 & -L_3\sin\psi \\ 1 & -L_2\cos\theta & 0 & -1 & L_3\cos\psi \\ 1 & -D\cos\theta & 0 & 0 & 0 \end{bmatrix}$