$$\begin{pmatrix} 1+2\beta\left(1-\cos\phi\right) & -\beta\left(1-\cos\phi\right) \\ \beta\left(1-\cos\phi\right) & -\beta \end{pmatrix} \begin{pmatrix} \ddot{\theta} \\ \ddot{\phi} \end{pmatrix}$$

$$+ \begin{pmatrix} -\beta\sin\phi\left(\dot{\phi}^2 - 2\dot{\theta}\dot{\phi}\right) \\ \beta\dot{\theta}^2\sin\phi \end{pmatrix}$$

$$+ \begin{pmatrix} \frac{\beta g}{l}\left[\sin\left(\theta-\phi-\gamma\right)-\sin\left(\theta-\gamma\right)\right] - \frac{g}{l}\sin\left(\theta-\gamma\right) \\ \frac{\beta g}{l}\sin\left(\theta-\phi-\gamma\right) \end{pmatrix} = 0$$

```python
def equations(t, variable_list):

    theta, theta_dot, phi, phi_dot = variable_list

    matrix_A = np.array([[1+2*beta*(1-np.cos(phi)), -beta*(1-np.cos(phi))],
                         [beta*(1-np.cos(phi)),     -beta                ]])

    vector_B = np.array([-beta*np.sin(phi)*(phi_dot**2-2*theta_dot*phi_dot),
                         beta*theta_dot**2*np.sin(phi)                     ])

    vector_C = np.array([beta*g/l*(np.sin(theta-phi-gamma)-np.sin(theta-gamma)-g/l*np.sin(theta-gamma)),
                         beta*g/l*np.sin(theta-phi-gamma)])

    theta_ddot, phi_ddot = np.linalg.solve(matrix_A, -vector_B-vector_C)

    return [theta_dot, theta_ddot, phi_dot, phi_ddot]
```

变量解包 $\theta,\ \dot{\theta},\ \phi,\ \dot{\phi}$

解出 $\ddot{\theta},\ \ddot{\phi}$

以列表的形式返回一次微分后的结果

之后会对它们积分