

# Equations de mouvement pour 3-omniwheels

Hypothèses :

- roues aux sommets d'un triangle équilatéral avec 120° entre elles
- chaque roue peut avoir un rayon différent
- chaque roue peut se trouver à une distance différente du centre du triangle équilatéral
- référentiel robot au centre du triangle équilatéral

## EQUATIONS DE MOUVEMENT ( MODEL DIRECT )

- $\omega_{t_i}$  = vitesses de rotation de la roue  $i$  pour la translation de la base holonome [rad/s]  
 $V$  = vitesse de déplacement du robot [m/s]  
 $r_i$  = rayon de la roue  $i$  [m]  
 $\theta_R$  = angle absolu de l'orientation du robot [rad]  
 $\theta_V$  = angle absolu de la direction de vitesse du robot [rad]  
 $\beta_i$  = angle relatif au robot de l'orientation de la roue  $i$  [rad]

$$\omega_{t_i} = \frac{V}{r_i} \cdot \cos\left(\theta_R - \theta_V + \beta_i - \frac{\pi}{2}\right)$$

```
beta1 = 60*pi/180;  
beta2 = 180*pi/180;  
beta3 = 300*pi/180;
```

```
omega1_t = linear_speed_value / r1 * cos(heading-speed_angle+beta1-pi/2);  
omega2_t = linear_speed_value / r2 * cos(heading-speed_angle+beta2-pi/2);  
omega3_t = linear_speed_value / r3 * cos(heading-speed_angle+beta3-pi/2);
```

- $\omega_{r_i}$  = vitesses de rotation de la roue  $i$  pour la rotation de la base holonome [rad/s]  
 $\Omega$  = vitesse de rotation du robot sur lui-même [rad/s]  
 $D_i$  = distance entre plan de rotation de la roue et centre de la base holonome [m]  
 $r_i$  = rayon de la roue  $i$  [m]

$$\omega_{r_i} = -\Omega \cdot \frac{D_i}{r_i}$$

```
omega1_r = -angular_speed_value * D1 / r1;  
omega2_r = -angular_speed_value * D2 / r2;  
omega3_r = -angular_speed_value * D3 / r3;
```

- $\omega_i$  = vitesse angulaire de la roue  $i$  [rad/s]

$$\omega_i = \omega_{t_i} + \omega_{r_i}$$

```
omega1 = omega1_t + omega1_r;
omega2 = omega2_t + omega2_r;
omega3 = omega3_t + omega3_r;
```

- $\alpha_i$  = positions angulaires de la roue  $i$  (facultatif) [rad]

$$\alpha_i = \alpha_i + \omega_i \cdot \Delta t$$

```
theta1 = theta1 + omega1*dt;
theta2 = theta2 + omega2*dt;
theta3 = theta3 + omega3*dt;
```

## EQUATIONS DE L'ODOMETRIE ( MODEL INVERSE )

- (obtention de la direction de déplacement depuis les derniers incréments codeurs)

```
heading = heading - (steps1*r1+steps2*r2+steps3*r3) / (D1+D2+D3) / steps_turn;
```

- (obtention du déplacement de translation depuis les derniers incréments codeurs)

```
mov_x = cos(beta1)*steps1*r1/steps_turn
      + cos(beta2)*steps2*r2/steps_turn
      + cos(beta3)*steps3*r3/steps_turn;
mov_y = sin(beta1)*steps1*r1/steps_turn
      + sin(beta2)*steps2*r2/steps_turn
      + sin(beta3)*steps3*r3/steps_turn;
```

```
mov_x = mov_x *2/3;
mov_y = mov_y *2/3;
```

- (application de la matrice de rotation pour avoir les positions “table”)

```
pos_x = pos_x + cos(heading-pi/2)*mov_x - sin(heading-pi/2)*mov_y;
pos_y = pos_y + sin(heading-pi/2)*mov_x + cos(heading-pi/2)*mov_y;
```