technische universität
dortmund

Master Thesis

Estimation of Underactuated Degrees of Freedom(DOF's) in Humanoid Robots

Rajesh Rajendran
rajesh.rajendran@tu-dortmund.de

August 31, 2013

Examiner:

Name of first examiner	Name of second examiner
Abteilung Informationstechnik	Lehrstuhl des Zweitgutachters
Institut für Roboterforschung	Fakultät für Informatik

CONTENTS

1

INTRODUCTION

1.1 Motivation

The field of *Robotics* have seen a tremendous development since the introduction of the term by *Isaac Asimov* in 1940s. The fundamental components of robotic systems are mechanical structure, actuators, sensors and controller. Robotic system ranges from simple *Cartesian manipulator* to the complex *Humanoids*. *Industrial robots* are robots that are used in applications such as palletizing, material loading and unloading, part sorting, packaging etc. These robots usually operate in the structured environment whose geometrical or physical characteristics are known in priori. They are pre programmed to execute the set of tasks. These robots have largely aided the automation of manufacturing processes in the industries. *Mobile robots* that are used in the environments where human beings can hardly survive or be exposed to unsustainable risks are called *Field robots*. *Field robots* normally operate in the unstructured environments, where the geometry or physical characteristics are not know in priori. Mars rover *Curiosity* is one such example. Locomotion in these robots are achieved either by wheels or by mechanical legs. Operating in the unknown environments and dynamic balancing of mechanical structure demands advanced control schemes for *Field robots*.

Figure 1.1: Eine Beispiel-Abbildung.

2

STATE ESTIMATION

State estimation is the principle of estimating the internal state of the system from the measurement of inputs and outputs of the system. In general knowledge of the internal state of the system will make the system easy to control. Figure 2.1 shows the usage of state estimator in state feedback control loop.

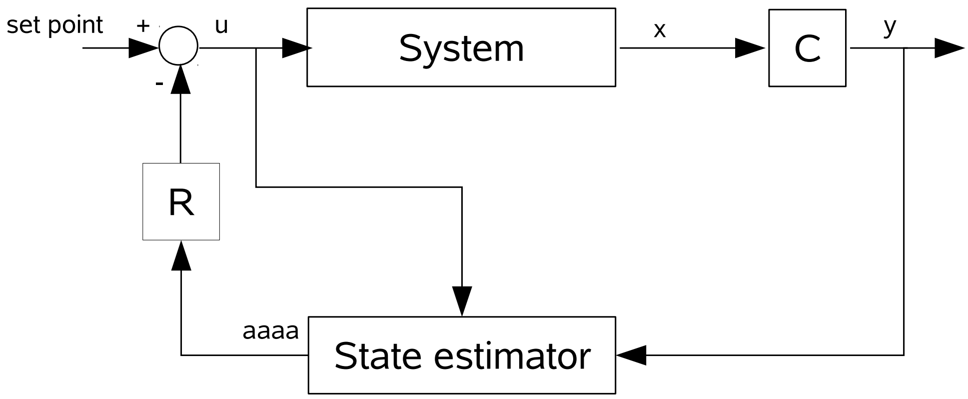

Figure 2.1: Structure of state feedback controller with state estimator

A general nonlinear system in state space form,

$$
\begin{aligned}
\dot{x}(t) &= f(x(t), u(t)), x(t=0) = x_0 \\
y(t) &= g(x(t), u(t))
\end{aligned}
\tag{2.1}
$$

In Equation 2.1, *x(t)* represents the vector of internal states, *u(t)* represents the vector of inputs and *y(t)* represents the vector of outputs of the system. x_0 is the initial state of the system which is usually unknown. The state estimator is described by the system equation with additional correction term

$$\begin{aligned}\dot{\hat{x}}(t) &= f(\hat{x}(t), u(t)) + K(y(t) - \hat{y}(t)), \hat{x}(t=0) = \hat{x}_0 \\ \hat{y}(t) &= g(\hat{x}(t), u(t))\end{aligned} \tag{2.2}$$

$\hat{x}(t)$ is the state vector of estimator and K is the gain matrix. A state estimator should satisfy the following properties

- **Simulation property:** For the same initial condition $x(t_0) = \hat{x}_0$ of the estimator and the system to be observed, then it holds that $x(t) = \hat{x}(t) \forall t > 0$.
- **Convergence property:** If $x(t_0) \neq \hat{x}_0$, then $x(t) - \hat{x}(t)$ tends to zero as $t \to \infty$

The different approaches for state estimator design differs in the calculation of gain matrix K in Equation 2.2.

2.1 Kalman Filter

Kalman filter is a statistical state estimation algorithm which estimates the internal state of the system from the noisy measurements. It was designed by Rudolph E. Kalman in 1960 for discrete time linear systems. It is basically a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance. Since the measurements occur and the states are estimated at discrete points of time, it is easily implementable in digital computers. Kalman filters are extensively used in the area of autonomous and guided navigation.

2.1.1 Kalman gain

Given a discrete time linear system affected by random noise

$$\begin{aligned}x_k &= Ax_{k-1} + Bu_k + w_{k-1} \\ y_k &= Hx_k + v_k\end{aligned} \tag{2.3}$$

where the random variables w_k,v_k represent the process and measurement noise. Both the random variables are assumed to be zero mean Gaussian white noises. Let Q, R be the covariance of process and measurement noise. Let us assume,

$$e_k^- = x_k - \hat{x}_k^- \tag{2.4}$$

be the error between the actual and predicted value of the state. The error covariance is given by

$$P_k^- = E[e_k^- e_k^{-T}] \tag{2.5}$$

Kalman filter corrects it estimate based on the predicted state and measured output data by

$$\hat{x_k} = \hat{x_k}^- + K(y_k - H\hat{x_k}^-) \tag{2.6}$$

Kalman gain is computed by substituting Equation 2.6 in Equation 2.4 to compute the e_k^-. Computed e_k^- is substituted in Equation 2.5 and the expected values are computed to find the error covariance P_k^-. Finally K is computed by taking the derivative of trace of P_k^- and equating it to zero

$$\frac{\partial trace(P_k^-)}{\partial K} = 0$$

solving the above equation for K. One form of K that minimizes Equation 2.6

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \tag{2.7}$$

From the Equation 2.7 as measurement covariance R approaches zero, Kalman gain K lays more trust on actual measurement y_k. On the other hand if P_k^- approaches zero, predicted measurement $H\hat{x_k}^-$ is trusted more.

2.1.2 Extended Kalman filter

Most of the real world estimation scenarios are non linear in nature. Kalman filter algorithm cannot be applied to the non linear systems. *NASA Ames* devised a method to apply Kalman filter for non linear systems which is called the Extended Kalman filter(EKF). In EKF the non linear system is linearised by multivariate Taylor series expansion of the non linear function.

Given a discrete time non linear system,

$$\begin{aligned} x_k &= f(x_{k-1}, u_k, w_{k-1}) \\ y_k &= h(x_k, u_k, v_k) \end{aligned} \tag{2.8}$$

x,y denotes the vector of system's state and output. *w,v* represents the process and measurement covariance noise. f is the non linear function that relates the previous state to the current state and h is the non linear function that relates the output and state.

In practice the individual values of noise w_k and v_k at each time step k is not known. So one can compute the approximated state and measurement vector without them as

$$\begin{aligned} \hat{x}_k^- &= f(\hat{x}_{k-1}, u_k, 0) \\ \hat{y}_k^- &= h(\hat{x}_k^-, u_k, 0) \end{aligned} \tag{2.9}$$

$\hat{x}_k^-$ and $\hat{y}_k^-$ are the *priori* estimates of state and measurements at time step k computed from *posteriori* estimate of state $\hat{x}_{k-1}$ from previous time step *k-1*.

A_k and H_k be the Jacobian matrices that results taking partial derivative of f and h with respect to x at time instant k. W_k and V_k be the Jacobian matrices that results taking partial derivative of f with respect to w and h with respect to v at time step k.

$$\begin{aligned}
A_k(i,j) &= \frac{\partial f_i}{\partial x_j}(\hat{x}_{k-1}, u_k, 0) \\
C_k(i,j) &= \frac{\partial h_i}{\partial x_j}(\hat{x}_k^-, u_k, 0) \\
W_k(i,j) &= \frac{\partial f_i}{\partial w_j}(\hat{x}_{k-1}, u_k, 0) \\
V_k(i,j) &= \frac{\partial h_i}{\partial v_j}(\hat{x}_k^-, u_k, 0)
\end{aligned} \tag{2.10}$$

At each time step these Jacobian matrices are evaluated with current predicted states $\hat{x}_k^-$.

2.1.2.1 Algorithm

Predict

$$\begin{aligned}
\hat{x}_k^- &= f(\hat{x}_{k-1}, u_k, 0) \\
P_k^- &= A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T
\end{aligned} \tag{2.11}$$

Correct

$$\begin{aligned}
K_k &= P_k^- H^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1} \\
\hat{x}_k &= \hat{x}_k^- + K_k(y_k - h(\hat{x}_k^-, u_k, 0)) \\
P_k &= (I - K_k H_k) P_k^-
\end{aligned} \tag{2.12}$$

2.1.3 Unscented Kalman filter

3

Multi body system model

This chapter discusses two modelling approaches used in the Kalman filter. In the first approach, the robot is modelled as multi body dynamic system in which the motion of the robot is caused by application of torques at the respective joints. In the second approach a sensor fusion algorithm is implemented which depends on the motion equations developed from a simplified model of a inertial navigation system.

3.1 Multi body system

Humanoid robots consists of group of rigid bodies that are joined together to resemble the human skeletal structure. As human beings are driven by the muscular system, humanoids are driven by electric, hydraulic or pneumatic actuators, which apply torques at the joints of the robot. The dynamics of robots describes how the robots moves in response to the actuator forces or torques. The dynamics of the robot is defined by the equations of motion of the rigid body system. In general there are several approaches in deriving equations of motion of rigid body dynamic system. One such approach is by deriving Lagrange equation. Lagrangian analysis relies on the energy properties of the mechanical system to derive the equations of motion. The *Lagrangian*, L, is defined as the difference between kinetic and potential energy of the system.

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q)$$

q represents the vector of joint angles T and V represents the kinetic and potential energies of the system. q is called the generalized coordinates of the system as the specification of joint angles uniquely determines position of all the rigid bodies.

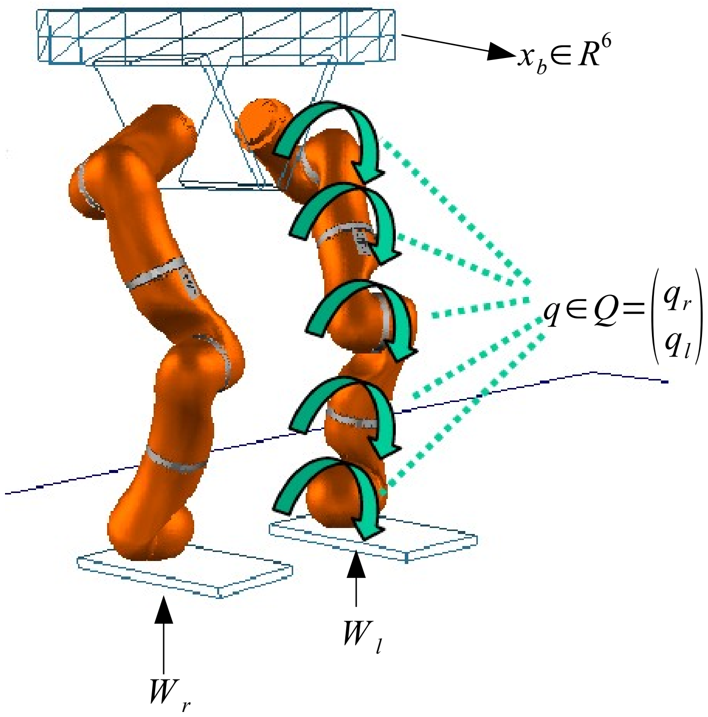

Figure 3.1: Simplified lower body model of biped

The equations of motion for a rigid body system with generalized coordinates $q \in \Re^m$ is given by,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - s\frac{\partial L}{\partial q_i} = \Gamma_i \qquad\qquad i = 1, ..., m \tag{3.1}$$

Γ_i is the external force acting on the i^{th} generalized coordinate. The general formulation of equation of motion of a multi body system is

$$M(q)\ddot{q} + C(q, \dot{q}) + g(q) = \tau + \tau_{ext} \tag{3.2}$$

M(q) is the generalized inertia matrix, $C(q, \dot{q})$ is the matrix representing Coriolis and centrifugal forces, *g(q)* is the gravity vector acting on the system and τ is the actuator torque applied to the joints τ_{ext} is the external torque acting on the system.
Figure 3.1 shows a biped with two legs, which is connected to common base. q represents a vector of joint variables which corresponds to angle of the joint. q_r, q_l corresponds to joins in left and right legs respectively. Number of joints corresponds to the number of controllable degrees of freedom of the robot. Q is the configuration space of the biped. x_b represents the number of degrees of freedom of the base. A rigid body in space has six degrees of freedom as shown in Figure 3.2 where *x,y,z* are the translational degrees of freedom and *roll,pitch,yaw* are the rotational degrees of freedom which corresponds

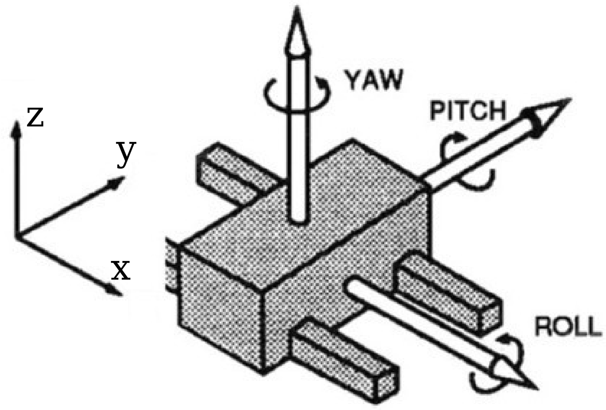

Figure 3.2: Degrees of freedom of a rigid body [1]

to rotation around *x,y,z* axes respectively. In biped the base is a free rigid body which is unactuated. The base has all six degrees of freedom unless it is constrained to the surface as shown in Figure 3.1. Since the base has all the six degrees of freedom it is called *floating base* (i.e it is considered as floating in space). The equation of motion of the floating base is given by Newton-Euler equation of motion in body coordinates **(author?)** [MLS94, chapter 4].

$$\begin{bmatrix} mI & 0 \\ 0 & \Im \end{bmatrix} \begin{bmatrix} \dot{v}^b \\ \dot{\omega}^b \end{bmatrix} + \begin{bmatrix} \omega^b \times mv^b \\ \omega^b \times \Im w^b \end{bmatrix} = W^b \tag{3.3}$$

m is the mass of the rigid body, $\Im$ is the inertia of the rigid body. $I, 0$ are the 3×3 identity and zero matrices. $[\dot{v}^b, \dot{\omega}^b]$ are the body twist coordinates of the rigid body. W^b is the body wrench applied to the center of mass of body.

$$M_x^b \ddot{x}_f^b + C_x^b \dot{x}_f^b + g_x^b = W^b \tag{3.4}$$

To be consistent with the multi body system dynamics formulation in Equation 3.2 Newton Euler Equation 3.3 is reformulated into Equation 3.4. x_f^b is a vector that represents six degrees of freedom of the rigid body. $\dot{x}_f^b$ represents the body twist. $\ddot{x}_f^b$ represents the acceleration in body frame(i.e it represents the time derivative of body twist). M_x^b represents the inertia matrix of the rigid body in body coordinates. C_x^b is the matrix representing the Coriolis and centrifugal forces acting on the system in body coordinates. g_x^b is the vector representing the gravitational forces acting on the body in body coordinates. W^b is the external force applied to the center of mass of the body. [2].

[1]Image source:`http://www.cncexpo.com/Images/pitchyawroll.jpg`

[2]Inertia, Coriolis and gravity are assumed to be given with respect to the body coordinate frame for the rest of the report

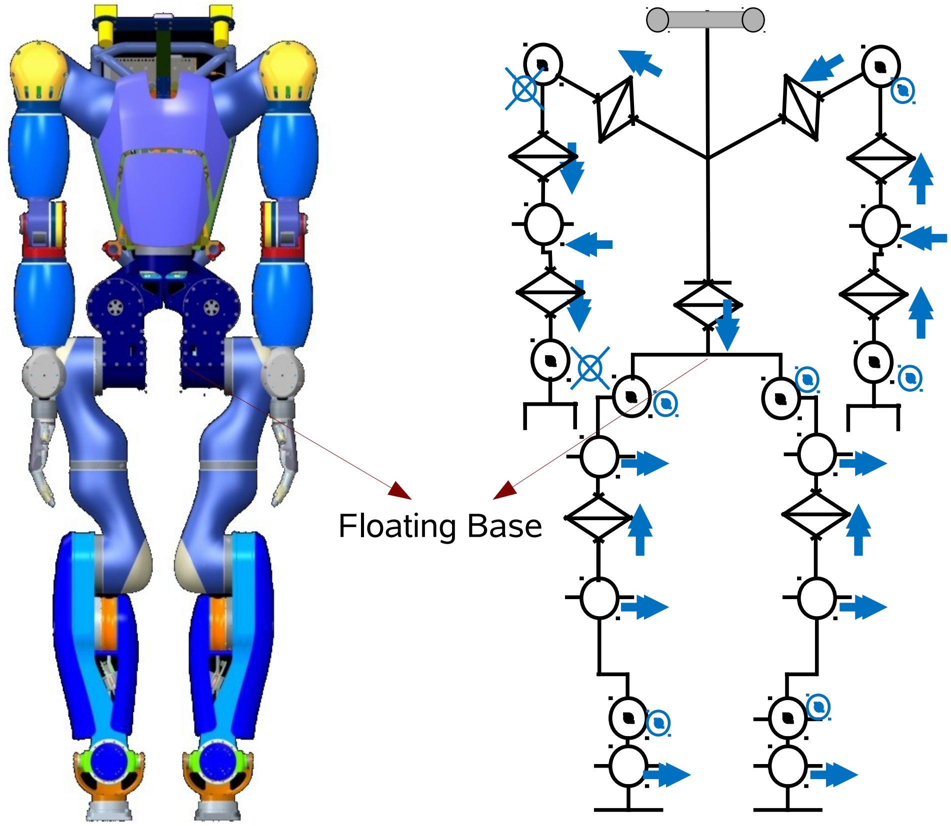

Figure 3.3: Kinematic chain of *Toro* Explain the joints symbols

The dynamics of biped in Figure 3.1 is composed of both multi body dynamics in the form of legs and rigid body dynamics in form of base. Combining Equation 3.4 and Equation 3.2 gives the equation of the biped.

$$\begin{bmatrix} M_x & M_{xq} \\ M_{qx} & M_q \end{bmatrix} \begin{bmatrix} \ddot{x}_f \\ \ddot{q} \end{bmatrix} + \begin{bmatrix} C_x & C_{xq} \\ C_{qx} & C_q \end{bmatrix} \begin{bmatrix} \dot{x}_f \\ \dot{q} \end{bmatrix} + \begin{bmatrix} g_x \\ g_q \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix} + (J_r^b)^T W_r^b + (J_l^b)^T W_l^b \tag{3.5}$$

Explanation of the terms in the equation are part of appendix. Equation 3.5 can be written in a simplified form as

$$M(y)\ddot{y} + C(y,\dot{y})\dot{y} + g(y) = W_{act} + W_{ext} \tag{3.6}$$

where $y = [x_f, q]^T$ is the combined state variables of the biped. $M(y), C(y), g(y)$ are combined Inertia, Coriolis and gravity parameters of the biped.

Toro is modelled as floating base dynamic model. *Toro* is made up of 25 rotary joints that are actuated by electrical motors. It can be seen from Figure 3.3 that floating base acts as the root of the kinematic chain. Legs and torso branches out of the floating base. Equation of motion for *Toro* can be formulated in similar fashion as biped Equation 3.6, where the dynamics of the upper body is also combined to it. The forward dynamics of *Toro*

$$
\begin{aligned}
\ddot{y} &= M(y)^{-1}(-C(y,\dot{y})\dot{y} - g(y) + J_r(y)^T W_r + J_l(y)^T W_l + \tau) \\
&\text{where,} \\
y &= \begin{bmatrix} x_f \\ q \end{bmatrix} \in \Re^{31} \\
x_f &= [p,\theta]^T \in \Re^6 \\
q &= [q_1, q_2, \ldots, q_{25}]^T \in \Re^{25}
\end{aligned} \tag{3.7}
$$

q is a vector of joint variables given in generalized coordinates. $x_f = [p,\theta]^T$ is a vector of position and orientation defined with respect to spatial frame. $p = [p_x, p_y, p_z]$ is a vector representing the position of origin of floating base with respect to spatial frame and $\theta = [\theta_x, \theta_y, \theta_z]$ is a vector of Euler angles that describes the rotation of floating base frame with respect to spatial frame. $q \in \Re^{25}$ is the vector of joint angles. $\dot{y} = [V^b, \dot{q}]^T \in \Re^{31}$ is the vector of generalized velocities. $V^b = [v^b, \omega^b]^T \in \Re^6$ is the body velocity, it is defined in 3.9. $\dot{q} \in \Re^{25}$ is the vector of joint velocities. $\ddot{y} \in \Re^{31}$ is the vector of generalized accelerations. $M(y) \in \Re^{31\times 31}$ is the inertia matrix, $C(y,\dot{y}) \in \Re^{31\times 31}$ is the matrix accounting for centrifugal and Coriolis forces. $g(y) \in \Re^{31}$ is the gravity vector. $\tau \in \Re^{31}$ is the vector of actuating torques acting on the robot, where the first six components are zero because those degrees of freedom corresponding to x_f are not actuated. $J_r(y)^T, J_l(y)^T \in \Re^{31\times 6}$ represents the body Jacobian that transforms the wrenches $W_r, W_l \in \Re^6$ applied in the right and left foot to generalized forces acting on the robot.

Integrating the forward dynamics Equation 3.7 once gives the velocity of all the bodies in the system and integrating it twice gives the position or orientation of the system.

$$acc = \frac{\partial^2 pos}{\partial t}, vel = \frac{\partial pos}{\partial t}$$

3.1.1 Space representation:

The multi body system would be modeled with position(*pos*) and velocity(*vel*) as the state variables which leads to following form

$$\begin{bmatrix} \dot{pos} \\ \dot{vel} \end{bmatrix} = \begin{bmatrix} vel \\ acc \end{bmatrix} \tag{3.8}$$

acceleration(*acc*) is given by the forward dynamics of the system.

The equations of motion in Equation 3.7 should be formulated in state space form of non linear systems as given in 2.1. The velocity of the free floating base of the robot is modelled as body velocity V_f^b which is given as,

$$V^b = \begin{bmatrix} v^b \\ \omega^b \end{bmatrix} = \begin{bmatrix} R^T \dot{p} \\ (R^T \dot{R})^{\vee} \end{bmatrix} \tag{3.9}$$

$\vee$ operator denotes represents the extraction of 3 dimensional vector from the symmetric matrix[Appendix 5.4] $\dot{p}$ is the time rate of change of the position of the body. R is the rotation matrix which describes the rotation of rigid body with respect to spatial frame. It is possible to reformulate the translational velocity of the above equation in first order ODE(Ordinary Differential Equation). It is not straight forward to obtain the time rate of change of Euler angles $\dot{\theta}$. There exists a transformation between the $\dot{\theta}$ and angular velocity ω^b.

$$\omega^b = T(\theta)\dot{\theta} \tag{3.10}$$

$T(\theta)$ is the transformation matrix [Appendix 5.4].
The state space representation of the equation of motions of *Toro* can be obtained by substituting Equations 3.9, 3.10 and 3.7 in Equation 3.8

$$\dot{x} = \begin{bmatrix} \dot{p} \\ \dot{\theta} \\ \dot{q} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} Rv^b \\ T(\theta)^{-1}\omega_f^b \\ \dot{q} \\ M(y)^{-1}(-C(y,\dot{y})\dot{y} - g(y) + J_r(y)^T W_r + J_l(y)^T W_l + \tau) \end{bmatrix} \tag{3.11}$$

- $$y = \begin{bmatrix} p \\ \theta \\ q \end{bmatrix} = \begin{bmatrix} x_f \\ q \end{bmatrix}, \dot{y} = \begin{bmatrix} V^b \\ \dot{q} \end{bmatrix}, x = \begin{bmatrix} y \\ \dot{y} \end{bmatrix}$$

  x_f, q are the parameters of the floating base and joints as described in 3.7. V^b is the body velocity as defined in 3.9 and $\dot{q}$ is the velocities of the joints of the robot. x is the vector of system states.

- $T(\theta_f)$ is the matrix that transforms the angular velocity ω_f^b to the time derivative of Euler angles $\dot{\theta}_f$. i.e $\omega_f^b = T(\theta_f)\dot{\theta}_f$.

- R is the rotation matrix which describes the rotation of floating base with respect to spatial frame. $R = R_x(\theta_x)R_y(\theta_y)R_z(\theta_z)$

3.2 Prediction Step

The prediction equations of EKF are given in Equation 2.11. For the sake of simplicity we assume the process noise acting on the model is uncorrelated. i.e The noise acting on each state is independent

$$W_k = I_3$$

. Substituting the value of W_k in Equation 2.11

$$\begin{aligned} \hat{x}_{k+1}^- &= f(\hat{x}_k, u_{k+1}, 0) \\ P_{k+1}^- &= A_k P_k A_k^T + Q_k \end{aligned} \tag{3.12}$$

The model is discretized for the implementation of EKF. Since the time step for integration is very small $\Delta t = 1ms$ forward Euler discretization method is used to discretize the continuous time model in 3.11.

$$\begin{bmatrix} \hat{p}_{k+1}^- \\ \hat{\theta}_{k+1}^- \\ \hat{q}_{k+1}^- \\ \hat{y}_{k+1}^- \end{bmatrix} = \begin{bmatrix} \hat{p}_k \\ \hat{\theta}_k \\ \hat{q}_k \\ \hat{y}_k \end{bmatrix} + \Delta t f(\hat{x}_k, u_{k+1}) \tag{3.13}$$

$$f(\hat{x}_k, u_{k+1}) = \begin{bmatrix} Rv_k^b \\ T(\hat{\theta}_k)^{-1}\omega_k^b \\ \dot{q}_k \\ M(\hat{y}_k)^{-1}(-C(\hat{y}_k, \hat{y}_k)\hat{y}_k - g(\hat{y}_k) + J_r(\hat{y}_k)^T W_{r,k+1} + J_l(\hat{y}_k)^T W_{l,k+1} + \tau_{k+1}) \end{bmatrix}$$

$\hat{x}(t_k) = \hat{x}(k\Delta t) = \hat{x}_k$ represents the state x at kth sampling instant. $\hat{x}_{k+1} = \hat{x}(k\Delta t + \Delta t)$ represents the state of the system at the next sampling instant. u_{k+1} is the input at sampling instant $k+1$. It is assumed that the value of the input remains constant in the interval between two sampling instant. This assumption is valid because of the zero order hold mechanism in sensor circuitry.

Equation 3.13 is used to predict the state $\hat{x}_k$ in Equation 3.12. For the computation of state covariance matrix P_k^- in Equation 3.12, the Jacobian Matrix is computed for Equation 3.13. The Jacobian computation of the different parts of the equation is follows, From Equation 3.13

1. $\hat{p}_{k+1}^- = \hat{p}_k + \Delta t R v_b$, $\hat{p}_k = [\hat{p}_{x,k}, \hat{p}_{y,k}, \hat{p}_{z,k}]$

$$\frac{\partial \hat{p}_{j,k+1}^-}{\partial x} = \left(\frac{\partial \hat{p}_{j,k+1}^-}{\partial x_1}, \frac{\partial \hat{p}_{j,k+1}^-}{\partial x_2}, \cdots, \frac{\partial \hat{p}_{j,k+1}^-}{\partial x_{62}} \right) \in \Re^{3\times 62} \tag{3.14}$$

$$\frac{\partial \hat{p}_{k+1}^{-}}{\partial x_i} = \begin{cases} e_i & \text{if } (i = j) \\ \Delta t \dfrac{\partial R}{\partial x_i} v_b & \text{if } (3 < i \leq 6) \\ \mathbf{0}_{3\times 1} & \text{if} (6 < i \leq 31) \text{ or } (35 < i \leq 62) \\ col(R, i-31) & \text{if } 31 < i \leq 34 \end{cases}$$

- j in the subscript represents the row dimension and i represents the column dimension of the matrix in Equation 3.14
- $col(X, i)$ - represents the *ith* column of matrix X.
- $\frac{\partial R}{\partial x_i}$ is the partial derivative of R with respect to the state $\hat{x}_k$ (i.e euler angles [Appendix 5.3]), e_i is the unit vectors in direction of coordinate axis and $\mathbf{0}_{3\times 1}$ is the zero vector of dimensions 3×1 [Appendix 5.1].

2. $\hat{\theta}_{k+1}^{-} = \hat{\theta}_k + \Delta t T(\hat{\theta}_k)^{-1} \omega_k^b$

$$\frac{\partial \hat{\theta}_{j,k+1}^{-}}{\partial x} = \left(\frac{\partial \hat{\theta}_{j,k+1}^{-}}{\partial x_1}, \frac{\partial \hat{\theta}_{j,k+1}^{-}}{\partial x_2}, \cdots, \frac{\partial \hat{\theta}_{j,k+1}^{-}}{\partial x_{62}} \right) \in \Re^{3\times 62} \tag{3.15}$$

$$\frac{\partial \hat{\theta}_{k+1}^{-}}{\partial x_i} = \begin{cases} \mathbf{0}_{3\times 1} & \text{if} (0 < i \leq 3) \text{ or } (6 < i \leq 31) \text{ or } (35 < i \leq 62) \\ e_{i-3} + \Delta t \dfrac{\partial T(\hat{\theta}_k^{-})^{-1}}{\partial x_i} \omega_k^b & \text{if3 < i} \leq 6 \\ col(T(\hat{\theta}_k^{-})^{-1}, i-34) & \text{if } 31 < i \leq 34 \end{cases}$$

- $\frac{\partial T(\theta_k^{-})^{-1}}{\partial x_i}$ is the partial derivative of inverse of transformation matrix with respect to state [Appendix 5.4]

3. $\hat{q}_{k+1}^{-} = \hat{q}_k + \Delta t \dot{q}_k$

$$\frac{\partial \hat{q}_{k+1}^{-}}{\partial x} = \left(\frac{\partial \hat{q}_{k+1}^{-}}{\partial x_1}, \frac{\partial \hat{q}_{k+1}^{-}}{\partial x_2}, \cdots, \frac{\partial \hat{q}_{k+1}^{-}}{\partial x_{62}} \right) \in \Re^{25\times 62} \tag{3.16}$$

$$\frac{\partial \hat{q}_{k+1}^{-}}{\partial x_i} = \begin{cases} l_{25,i} & \text{if } (6 < i \leq 31) \text{ or } (38 < i \leq 62) \\ 0 & \text{otherwise} \end{cases}$$

- $l_{25,i}$ is a colomn vector of length 25 with 1 in the *ith* position and zeros in other position [Appendix 5.1].

4. $\hat{y}_{k+1}^{-} = \hat{y}_k + \Delta t \Lambda$

$$\Lambda = M(\hat{y}_k)^{-1}(-C(\hat{y}_k, \hat{y}_k)\hat{y}_k - g(\hat{y}_k) + J_r(\hat{y}_k)^T W_{r,k+1} + J_l(\hat{y}_k)^T W_{l,k+1} + \tau_{k+1})$$

$$\frac{\partial \hat{y}_{k+1}^{-}}{\partial x} = \left(\frac{\partial \hat{y}_{k+1}^{-}}{\partial x_1}, \frac{\partial \hat{y}_{k+1}^{-}}{\partial x_2}, \cdots, \frac{\partial \hat{y}_{k+1}^{-}}{\partial x_{62}} \right) \in \Re^{31\times 62} \tag{3.17}$$

where,

$$\frac{\partial \hat{y}_{k+1}^-}{\partial x_i} = \begin{cases} \left. \begin{array}{c} -M_k^{-1}\frac{\partial M_k}{\partial x_i}\Lambda + M_k^{-1}\left(-\frac{\partial C_k}{\partial x_i}\hat{y}_k - \frac{\partial g_k}{\partial x_i} + \left(\frac{\partial J_{r,k}^b}{\partial x_i}\right)^T W_{r,k+1}\right) \\ + M_k^{-1}\left(\frac{\partial J_{l,k}^b}{\partial x_i}\right)^T W_{l,k+1} \end{array} \right\} & \text{if } 0 < i \leq 31 \\ l_{31,(i-31)} - M_k^{-1}\frac{\partial M_k}{\partial x_i}\Lambda + M_k^{-1}\left(-\frac{\partial C_k}{\partial x_i}\hat{y}_k - col(C_k, i-31)\right) & \text{if } i < 31 \end{cases}$$

- $M_k = M(\hat{y}_k), C_k = C(\hat{y}_k, \hat{y}_k), J_{r,k} = J_r(\hat{y}_k), J_{l,k} = J_l(\hat{y}_k)$

The system matrix A_k in Equation 3.12 is formulated by combining Equations 3.14,3.15,3.16 and 3.17

$$A_k = \begin{pmatrix} \frac{\partial \hat{p}_{k+1}^-}{\partial x} \\ \frac{\partial \hat{\theta}_{k+1}^-}{\partial x} \\ \frac{\partial \hat{q}_{k+1}^-}{\partial x} \\ \frac{\partial \hat{y}_{k+1}^-}{\partial x} \end{pmatrix} \in \Re^{62\times 62} \tag{3.18}$$

Substituting Equations 3.13 and 3.18 in 3.12 and substituting the values of process covariance Q_k completes the prediction stage of EKF.

3.3 Update Step

The update equation of the EKF is given in Equation 2.12. The measurement equation of the system is given by

$$\hat{y}_{k+1} = h(\hat{x}_{k+1}^-, u_{k+1}, 0)$$

.For the sake of simplicity let us assume the measurement of noise are independent.

$$V_k = I_3$$

. Substituting the assumption in 2.12

$$\begin{aligned} K_{k+1} &= P_{k+1}^- H_{k+1}^T (H_{k+1} P_{k+1}^- H_{k+1}^T + R_{k+1})^{-1} \\ \hat{x}_{k+1} &= \hat{x}_{k+1}^- + K_{k+1}(y_{k+1} - \hat{y}_{k+1}) \\ P_{k+1} &= (I - K_{k+1} H_{k+1}) P_{k+1}^- \end{aligned} \tag{3.19}$$

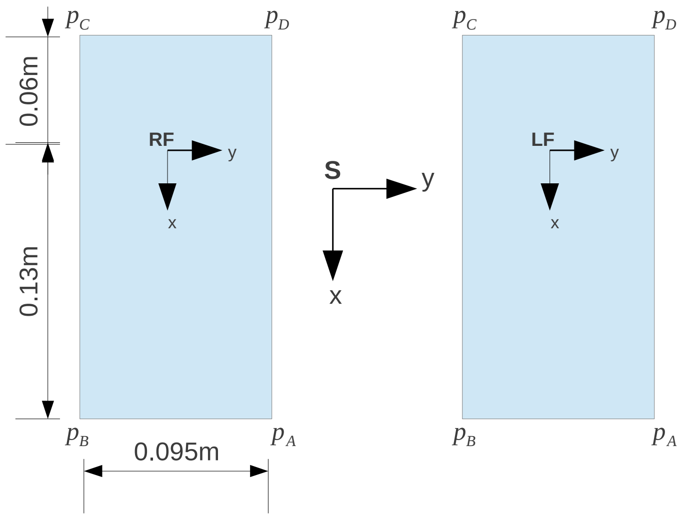

Figure 3.4: Toro feet viewed from top

The measurements of *Toro* are Cartisian accelerations(acc^b) of the hip measured by IMU, angular velocity(ω^b) of the hip measured by the gyroscope, joint angles(q_j) and joint velocities($\dot{q}_j$) measured by joint encoders.

$$y_{sens} = \begin{bmatrix} acc^b \\ \omega^b \\ q_j \\ \dot{q}_j \end{bmatrix} \tag{3.20}$$

- The simplified model of IMU is

$$acc^b = \begin{bmatrix} acc_x^b \\ acc_y^b \\ acc_z^b \end{bmatrix} = \begin{bmatrix} \ddot{p}_x^b \\ \ddot{p}_y^b \\ \ddot{p}_z^b \end{bmatrix} - R^T \begin{bmatrix} 0 \\ 0 \\ -9.81 \end{bmatrix}$$

  R is the rotational matrix that transforms a vector in body coordinate frame to spatial frame.

- $\ddot{p}^b$ is computed from the forward dynamic equation 3.7 using the predicted values of the state

- ω_f^b is the vector of angular rates of the hip(floating base) measured by gyroscope. The measurements are in the frame attached to the hip.

Along with the sensor measurements kinematic constraints are also considered as mesurement. When a foot is in contact with the ground the velocity of the foot is zero. Figure 3.4 shows the contact points considered for measurements. The corner points of each foot are measured with respect to spatial frame S as shown in Figure 3.4 before starting the experiment and they are assumed to be constant throughout the experiment. The contact points of the robot does not change throughout the experiment, since we are considering the case where the robot is tilting around one edge of the foot.

$$
\begin{aligned}
y_{kin} &= \begin{bmatrix} p_{contact} \\ V_{contact} \end{bmatrix} \\
p_{contact} &= \begin{bmatrix} p_{RF} \\ p_{LF} \end{bmatrix} \\
V_{contact} &= \begin{bmatrix} V^b_{RF} \\ V^b_{LF} \end{bmatrix} = \begin{bmatrix} J_r(\hat{y})^T \hat{\dot{q}} \\ J_l(\hat{y})^T \hat{\dot{q}} \end{bmatrix}
\end{aligned}
\tag{3.21}
$$

- p_{RF}, p_{LF} are the vectors of contact points in right foot and left foot defined with respect to spatial frame.
- $J_r(y), J_l(y)$ are the Jacobians of right and left foot that relates the joint velocity to the velocity of right and left foot respectively. [Appendix Define Body Jacobian]

$$
\begin{aligned}
p_{RF} &= \begin{bmatrix} p_{A,RF} \\ p_{B,RF} \\ p_{C,RF} \\ p_{D,RF} \end{bmatrix} = \begin{bmatrix} H_{RF} p_A \\ H_{RF} p_B \\ H_{RF} p_C \\ H_{RF} p_D \end{bmatrix} \\
p_{LF} &= \begin{bmatrix} p_{A,LF} \\ p_{B,LF} \\ p_{C,LF} \\ p_{D,LF} \end{bmatrix} = \begin{bmatrix} H_{LF} p_A \\ H_{LF} p_B \\ H_{LF} p_C \\ H_{LF} p_D \end{bmatrix}
\end{aligned}
\tag{3.22}
$$

- $\hat{H}_{RF}, \hat{H}_{LF}$ are the homogeneous transformation matrices of the right and left foot.Appendix Homogeneous Transformation Matrix
- In Figure 3.4 p_A, p_B, p_C, p_D are the corner points defined with respect to respective foot frame *RF,LF*.

The full measurement equations of is obtained combining Equations 3.20 and 3.21

$$
y_{k+1} = \begin{bmatrix} y_{sens,k+1} \\ y_{kin,k+1} \end{bmatrix} \tag{3.23}
$$

The measurement sensititvity matrix can be computed by taking the parial derivative of the mesaurement equation 3.23 with respect to system states x.

a) $a\hat{c}c^b_{k+1} = \ddot{p}_{k+1} - \hat{R}^T \begin{bmatrix} 0 \\ 0 \\ -9,81 \end{bmatrix}$

$$
\frac{\partial a\hat{c}c^b_{k+1}}{\partial x} = \frac{\partial \hat{\ddot{p}}^b_{k+1}}{\partial x} + \frac{\partial \hat{R}^T_{k+1}}{\partial x} \begin{bmatrix} 0 \\ 0 \\ -9,81 \end{bmatrix} \in \Re^{3\times 62} \tag{3.24}
$$

- $\frac{\partial \hat{p}^b_{k+1}}{\partial x}$ is parital derivative of acceleration of body with respect to states of the system. It is computed by substituting $\hat{x}^-_{k+1}$ for $\hat{x}_k$ in Equation 3.17 and then subtracting $l_{31,i-31}$ for the case $i > 31$. The first three rows of the reuslting matrix is the partial derivative of acceleration with respect to states.
- $\frac{\partial \hat{R}^T_{k+1}}{\partial x}$ is partial derivative of Rotation matrix with respect to system state.[Appendix 5.3]

b) $\hat{\omega}^{b-}_{k+1}$

$$\frac{\partial \hat{\omega}^{b-}_{k+1}}{\partial x} = \left(\frac{\partial \hat{\omega}^{b-}_{k+1}}{\partial x_1}, \frac{\partial \hat{\omega}^{b-}_{k+1}}{\partial x_2}, \cdots, \frac{\partial \hat{\omega}^{b-}_{k+1}}{\partial x_{62}} \right) \in \Re^{3\times 62} \tag{3.25}$$

$$\frac{\partial \hat{\omega}^{b-}_{k+1}}{\partial x} = \begin{cases} l_{3,34-i} & \text{if } 34 < i \leq 37 \\ \mathbf{0}_{3,1} & \text{otherwise} \end{cases}$$

c) $\hat{q}^-_{k+1}$

$$\frac{\partial \hat{q}^-_{k+1}}{\partial x} = \left(\frac{\partial \hat{q}^-_{k+1}}{\partial x_1}, \frac{\partial \hat{q}^-_{k+1}}{\partial x_2}, \cdots, \frac{\partial \hat{q}^-_{k+1}}{\partial x_{62}} \right) \in \Re^{25\times 62} \tag{3.26}$$

$$\frac{\partial \hat{q}^-_{k+1}}{\partial x_i} = \begin{cases} 1 & \text{if } 7 \leq i \leq 31 \\ 0 & \text{otherwise} \end{cases}$$

d) $\hat{\dot{q}}^-_{k+1}$

$$\frac{\partial \hat{\dot{q}}^-_{k+1}}{\partial x} = \left(\frac{\partial \hat{\dot{q}}^-_{k+1}}{\partial x_1}, \frac{\partial \hat{\dot{q}}^-_{k+1}}{\partial x_2}, \cdots, \frac{\partial \hat{\dot{q}}^-_{k+1}}{\partial x_{62}} \right) \in \Re^{25\times 62} \tag{3.27}$$

$$\frac{\partial \hat{\dot{q}}^-_{k+1}}{\partial x_i} = \begin{cases} 1 & \text{if } 37 < i \leq 62 \\ 0 & \text{otherwise} \end{cases}$$

$$\frac{\partial p^b_c}{\partial x} = \left(\frac{\partial p^b_c}{\partial x_1}, \frac{\partial p^b_c}{\partial x_2}, \cdots, \frac{\partial p^b_c}{\partial x_{62}} \right) \in \Re^{12\times 62} \tag{3.28}$$

$$\frac{\partial p^b_c}{\partial x_i} = \begin{cases} \begin{pmatrix} -H_{sr}^{-1} \dfrac{\partial H_{sr}}{\partial x_i} H_{sr}^{-1} p^s_{a,r} \\ -H_{sr}^{-1} \dfrac{\partial H_{sr}}{\partial x_i} H_{sr}^{-1} p^s_{b,r} \\ -H_{sl}^{-1} \dfrac{\partial H_{sl}}{\partial x_i} H_{sl}^{-1} p^s_{a,l} \\ -H_{sl}^{-1} \dfrac{\partial H_{sl}}{\partial x_i} H_{sl}^{-1} p^s_{b,l} \end{pmatrix} & \text{if } 1 \leq i \leq 31 \\ 0 & \text{otherwise} \end{cases}$$

$$\frac{\partial \hat{V}^b_c}{\partial x} = \left(\frac{\partial \hat{V}^b_c}{\partial x_1}, \frac{\partial \hat{V}^b_c}{\partial x_2}, \cdots, \frac{\partial \hat{V}^b_c}{\partial x_{62}} \right) \in \Re^{12\times 62} \tag{3.29}$$

$$\frac{\partial \hat{V}_c^b}{\partial x_i} = \begin{cases} \begin{pmatrix} \frac{\partial J_r^b}{\partial x_i}\dot{q} \\ \frac{\partial J_l^b}{\partial x_i}\dot{q} \end{pmatrix} & \text{if } 1 \leq i \leq 31 \\ \begin{pmatrix} col(J_r^b, i) \\ col(J_l^b, i) \end{pmatrix} & \text{if } 32 \leq i \leq 62 \end{cases}$$

The measurement matrix of the system is given by Eq. **??**, **??**, **??**, **??**, 3.28, 3.29

$$C = \begin{pmatrix} \frac{\partial q}{\partial x} \\ \frac{\partial \dot{q}}{\partial x} \\ \frac{\partial \ddot{p}_f^b}{\partial x} \\ \frac{\partial \dot{\theta}_f^b}{\partial x} \\ \frac{\partial p_c^b}{\partial x} \\ \frac{\partial \hat{V}_c^b}{\partial x} \end{pmatrix} \in \Re^{80\times 62} \tag{3.30}$$

Observability: State space representation of a linear system is,

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du. \end{aligned} \tag{3.31}$$

where, $x \in \Re^n$ is the vector representing the states of the system. $u \in \Re^p$ is the vector of inputs, $y \in \Re^m$ is the vector of outputs of the system. $A \in \Re^{n\times n}$ is the system matrix. $B \in \Re^{n\times p}$ is the matrix relating state and input, $C \in \Re^{m\times n}$ is the measurement matrix relating output and state, $D \in \Re^{m\times p}$ is the matrix relating input and output of the system. Linearising a non linear system in Equation 2.1 at some operating point will lead to linear system of form Eq. 3.31. For a linear system to be observable, it should satisfy

$$obs = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix}, rank(obs) = n \tag{3.32}$$

For our system to be observable $rank(obs) = 62$.

Make plots from files act=datsrc/ROBOT-TILT-0807.mat est=estimates-data/est-090701.mat or *080703.mat

4

SIMPLE MODEL

Inertial measurement unit IMU *MTi-100* [1] consists of accelerometer and gyroscope. The accelerometer measures accelerations a_x, a_y, a_z acting along the three Cartesian axes. Likewise gyroscope measures the angular rate $\omega_x, \omega_y, \omega_z$ along the respective axes. IMU measures the acceleration and angular rate with respect to the body frame with which it is attached. Normally the measurement from the IMU is accompanied by noise. In addition to the noise there can be bias acting on the measurements. The stochastic model of the IMU is

$$\begin{aligned}
\tilde{a} &= a + b_a + w_a \\
\dot{b}_a &= w_{ba} \\
\tilde{\omega} &= \omega + b_\omega + w_\omega \\
\dot{b}_\omega &= w_{b\omega}
\end{aligned} \tag{4.1}$$

$\tilde{a}$ is the measured acceleration from the IMU. It is composed of true acceleration a, bias in the acceleration measurements b_a and the sensor noise w_a. $\tilde{\omega}$ is the angular velocity measured, which comprises of true angular velocity ω, bias b_ω and sensor noise w_ω. The sensor noises w_a and w_ω are modelled as Gaussian white noise. The bias in acceleration measurement is b_a and that of angular velocity measurement is b_ω. The bias terms are also time varying quantities which follows slow dynamics. Bias dynamics $\dot{b}_a$ and $\dot{b}_\omega$ are modelled as first order Markov process. $w_{b\omega}$ and w_{bf} are the Gaussian white noise associated with the bias.

[1] xsens technologies `http://www.xsens.com/en/general/mti-100-series`

4.1 Prediction model

5

Appendix

5.1 Symbols

I_x	Identity matrix of dimension x
$\mathbf{0}_{r \times c}$	Zero matrix of dimensions given by r,c.
	r - row dimension , c - colomn dimension
e_i	Unit vectors pointing in direction of the coordinate axis
	$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ - unit vector along x axis
	$e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$ - unit vector along y axis
	$e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ - unit vector along z axis
$l_{m,n}$	Zero vector of length m, with 1 in nth position of the vector
	For Example, $l_{4,2} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$, $l_{2,1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

5.2 Abbreviation

EKF	Extended Kalman Filter
IMU	Inertial Measurement Unit
FTS	Force Torque sensor

5.3 Rotation matrix

The orientation of a rigid body in space can be represented in form of rotational matrix $R \in SO(3)$.

A 3 dimensional rotation matrix defines the rotation along *X,Y,Z* directions. The rotation matrix is composition of these individual rotations. The matrix representing the rotation around each coordinate axis is,

$$
\begin{aligned}
R_x(\theta_x) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta_x) & -sin(\theta_x) \\ 0 & sin(\theta_x) & cos(\theta_x) \end{bmatrix} \\
R_y(\theta_y) &= \begin{bmatrix} cos(\theta_y) & 0 & sin(\theta_y) \\ 0 & 1 & 0 \\ -sin(\theta_y) & 0 & cos(\theta_y) \end{bmatrix} \\
R_z(\theta_z) &= \begin{bmatrix} cos(\theta_z) & -sin(\theta_z) & 0 \\ sin(\theta_z) & cos(\theta_z) & 0) \\ 0 & 0 & 1 \end{bmatrix}
\end{aligned}
\tag{5.1}
$$

1. Rotation matrix for the whole body model in Chapter 3 is

$$
\begin{aligned}
R &= R_x(\theta_x)R_y(\theta_y)R_z(\theta_z) \\
&= \begin{bmatrix} c_y c_z & -c_y s_z & s_y \\ c_x s_z + c_z s_x s_y & c_x c_z - s_x s_y s_z & -c_y s_x \\ s_x s_z - c_x c_z s_y & c_z s_x - c_x s_y s_z & c_x c_y \end{bmatrix}
\end{aligned}
\tag{5.2}
$$

The partial derivative of the above rotational matrix with respect to the angles is

$$
\begin{aligned}
\frac{\partial R}{\partial \theta_x} &= \begin{bmatrix} 0 & 0 & 0 \\ c_x s_y c_z - s_x s_z & -c_x s_y s_z - s_x c_z & c_x c_y] \\ s_x s_y c_z + c_x s_z & -s_x s_y s_z + c_x c_z & -s_x c_y \end{bmatrix} \\
\frac{\partial R}{\partial \theta_y} &= \begin{bmatrix} -s_y c_z & s_y s_z & c_y \\ s_x c_y c_z & -s_x c_y s_z & s_x s_y \\ -c_x c_y c_z & c_x c_y s_z & -c_x s_y \end{bmatrix} \\
\frac{\partial R}{\partial \theta_z} &= \begin{bmatrix} -c_y s_z & -c_y c_z & 0 \\ -s_x s_y s_z + c_x c_z & -s_x s_y c_z - c_x s_z & 0 \\ c_x s_y s_z + s_x c_z & c_x s_y c_z - s_x s_z & 0 \end{bmatrix}
\end{aligned} \tag{5.3}
$$

2. Rotation matrix for the simplified model in Chapter 4is

$$
\begin{aligned}
R &= R_z(\theta_z) R_y(\theta_y) R_x(\theta_x) \\
&= \begin{bmatrix} c_z c_y & -s_z c_x + c_z s_y s_x & s_z s_x + c_z s_y c_x \\ s_z c_y & c_z c_x + s_z s_y s_x & -c_z s_x + s_z s_y c_x \\ -s_y & c_y s_x & c_y c_x \end{bmatrix}
\end{aligned} \tag{5.4}
$$

The partial derivative of the above rotational matrix with respect to the angles is

$$
\begin{aligned}
\frac{\partial R}{\partial \theta_x} &= \begin{bmatrix} 0 & s_z s_x + c_z s_y c_x & s_z c_x - c_z s_y s_x \\ 0 & -c_z s_x + s_z s_y c_x & -c_z c_x - s_z s_y s_x \\ 0 & c_y c_x & -c_y s_x \end{bmatrix} \\
\frac{\partial R}{\partial \theta_y} &= \begin{bmatrix} -c_z s_y & c_z c_y s_x & c_z c_y c_x \\ -s_z s_y & s_z c_y s_x & s_z c_y c_x \\ -c_y & -s_y s_x & -s_y c_x \end{bmatrix} \\
\frac{\partial R}{\partial \theta_z} &= \begin{bmatrix} -s_z c_y & -c_z c_x - s_z s_y s_x & c_z s_x - s_z s_y c_x \\ c_z c_y & -s_z c_x + c_z s_y s_x & s_z s_x + c_z s_y c_x \\ 0 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{5.5}
$$

- $\theta_x, \theta_y, \theta_z$ represents the rotation of the body coordinate frame relative to spatial frame coordinate frame.
- c_x, c_y, c_z are the short hand notations of $cos(\theta_x), cos(\theta_y), c_z$ respectively.
- s_x, s_y, s_z are the short hand notations of $sin(\theta_x), sin(\theta_y), sin(\theta_z)$ respectively.

5.4 Transformation of angular velocity to Euler rates

The relation between the angular velocity and time derivative of rotational matrix is

$$\omega^b = (R^T \dot{R})^\vee$$

- ω^b is the angular velocity of the body in represented in body coordinate frame.

- $R^T\dot{R}$ is a skew symmetric matrix [MLS94, Chapter 2].

$$R^T\dot{R} = \begin{bmatrix} 0 & -\omega_z^b & \omega_y^b \\ \omega_z^b & 0 & -\omega_x^b \\ -\omega_y^b & \omega_x^b & 0 \end{bmatrix}$$

- $\vee$ operator extracts the 3 dimensional vector that forms the skew symmetric matrix

$$(R^T\dot{R})^\vee = \begin{bmatrix} \omega_x^b \\ \omega_y^b \\ \omega_z^b \end{bmatrix} = w^b$$

The rotation matrix R is parametrized by Euler angles. w^b depends on the parametrization of the rotation matrix.

1. For the whole body model discussed in Chapter 3 the rotation matrix is given in Equation 5.2. The angular velocity is

$$\begin{aligned} \omega_x^b &= cos(\theta_z)cos(\theta_y)\dot{\theta}_x + sin(\theta_z)\dot{\theta}_y \\ \omega_y^b &= -sin(\theta_z)cos(\theta_y)\dot{\theta}_x + cos(\theta_z)\dot{\theta}_y \\ \omega_z^b &= sin(\theta_y)\dot{\theta}_x + \dot{\theta}_z \\ \begin{bmatrix} \omega_x^b \\ \omega_y^b \\ \omega_z^b \end{bmatrix} &= \begin{bmatrix} cos(\theta_z)cos(\theta_y) & sin(\theta_z) & 0 \\ -sin(\theta_z)cos(\theta_y) & c_z & 0 \\ sin(\theta_y) & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix} \\ \omega^b &= T(\theta)\dot{\theta} \end{aligned} \tag{5.6}$$

   $T(\theta)$ is the transformation matrix, that transforms the time derivative of euler angles parametrized in Equation 5.2.

   The partial derivatives of ω^b with respect to the states are

$$\frac{\partial \omega_b}{\partial \theta_x} = \mathbf{0}_{3\times3}, \frac{\partial \omega_b}{\partial \theta_y} = \begin{bmatrix} -c_z s_y & 0 & 0 \\ s_z s_y & 0 & 0 \\ c_y & 0 & 0 \end{bmatrix}, \frac{\partial \omega_b}{\partial \theta_z} = \begin{bmatrix} -s_z c_y & c_z & 0 \\ -c_z c_y & -s_z & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

2. For the simple model discussed in Chapter 4 the rotation matirx is given in Equation 5.4. The angular velocity is

$$\begin{aligned}
\omega_x^b &= \dot{\theta}_x - sin(\theta_y)\dot{\theta}_z \\
\omega_y^b &= cos(\theta_x)\dot{\theta}_y + cos(\theta_y)sin(\theta_x)\dot{\theta}_z \\
\omega_z^b &= -sin(\theta_x)\dot{\theta}_y + cos(\theta_y)cos(\theta_x)\dot{\theta}_z \\
\begin{bmatrix} \omega_x^b \\ \omega_y^b \\ \omega_z^b \end{bmatrix} &= \begin{bmatrix} 1 & 0 & -s_y \\ 0 & c_x & c_y s_x \\ 0 & -s_x & c_y c_x \end{bmatrix} \\
\omega^b &= T(\theta)\dot{\theta}
\end{aligned} \tag{5.7}$$

$T(\theta)$ is the transformation matrix, that transforms the time derivative of euler angles parametrized in Equation 5.4.

The partial derivatives of ω^b with respect to the states are

$$\frac{\partial \omega_b}{\partial \theta_y} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s_x & c_y c_x \\ 0 & -c_x & -c_y s_x \end{bmatrix}, \frac{\partial \omega_b}{\partial \theta_z} = \begin{bmatrix} 0 & 0 & -c_y \\ 0 & 0 & -s_y s_x \\ 0 & 0 & -s_y c_x \end{bmatrix}, \frac{\partial \omega_b}{\partial \theta_z} = \mathbf{0}_{3\times 3}$$

List of Figures

Bibliography

[MLS94] Murray, Richard M. ; Li, Zexiang ; Sastry, S.Shankar: *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994. – 474 S.

Hiermit bestätige ich, die vorliegende Diplomarbeit selbständig und nur unter Zuhilfenahme der angegebenen Literatur verfasst zu haben.

Ich bin damit einverstanden, dass Exemplare dieser Arbeit in den Bibliotheken der Universität Dortmund ausgestellt werden.

Dortmund, den August 31, 2013

Name