ME5402/EE5106 Advanced Robotics Part II

Project Report for

Mini Project 1-1 Modeling of the Robotic Dog

Song Qingfeng A0119083H

Tu Fangwen A0109982U

Wang Kangli A0055862Y

Abstract: This project report presents the research work done on the modeling of a robotic dog including literature review on quadruped robot dog design, kinematic and dynamic modeling of the robotic dog, and dynamic simulation and animation presentation. Based on the model, three different control laws are designed and simulation for the robotic dog leg control, namely, PD controller, computed torque method with LQR controller and adaptive controller. The results and finding are also presented.

Contents

1. Introduction

Animals like jumping lemurs, load hauling mules galloping cheetahs all conjure up highly desirable capabilities for machines in civilian, intelligence and military applications. While some animals can be trained to help out, machines with similar capabilities would be preferred. In addition, compared to animals, the load carrying machines could provide more functions to assist human beings such as power generation or water purification. Since most of animals' amazing terrestrial mobility is achieved with four legs, the quadruped robots development are highly motivated. With the grooming development of sensory technology, material science and control technic, the quadruped robots has been greatly developed recently. Some quite mature quadruped robots like *Bigdog*, *Wildcat* from Boston Dynamics and *AphaDog* from US Marines Robot Pack has shown their great terrestrial mobility and automotive ability.

In this project, we present the literature review on the basic design and implementation issue associated with the quadruped robots. Then we focus on the modeling of a simplified leg model of the quadruped robots. The kinematic and dynamic modeling are presented for the robot leg. Then based on the modeling, the PD control, computed torque method control and adaptive control are designed and simulated for trajectory tracking of the robotic leg. Simulation results and animation results are presented.

2. Literature Review

In the literature review part, three different quadruped robots are studied in details, which are *Bigdog* from Boston Dynamics[3] as shown in Figure 1, robot *Rush* from Tokyo University of Science [7] as shown in Figure 2, and *Cheetah-Cub* from Biorobotics Laboratory Biorob [4] as shown in Figure 3. These three robots are of different focus, different weights and different mobility. In this project, the mechanical design will be carefully studied and compared. The Table 1 shows a general comparison between the three robots.

Table 1 The general comparison between the three robots

Robot	Mass (kg)	Height(m)	Length(m)	Maximum Speed(ms^{-1})	FR	Gait
BigDog	109	1	1.1	3.1	0.98	Bound
Rush	4.3	0.2	0.3	0.9	0.41	Bound
Cheetah-cub	1.1	0.158	0.205	1.42	1.30	trot

Figure 1. Robot *Bigdog*

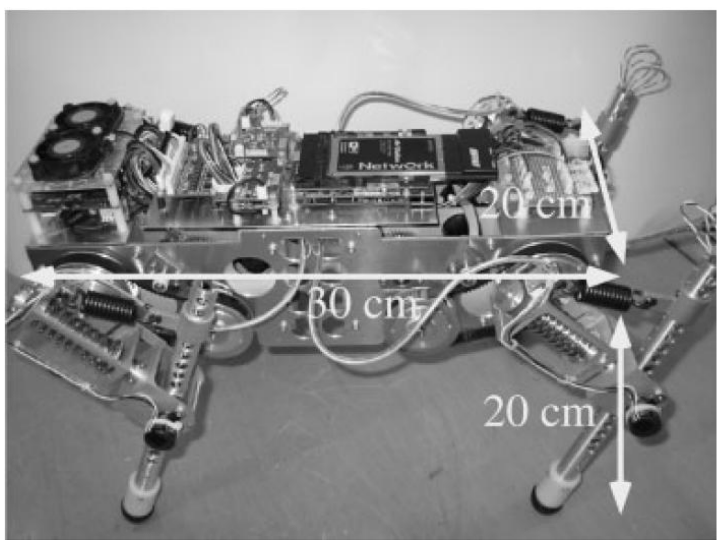


Figure 2. Robot *Rush*

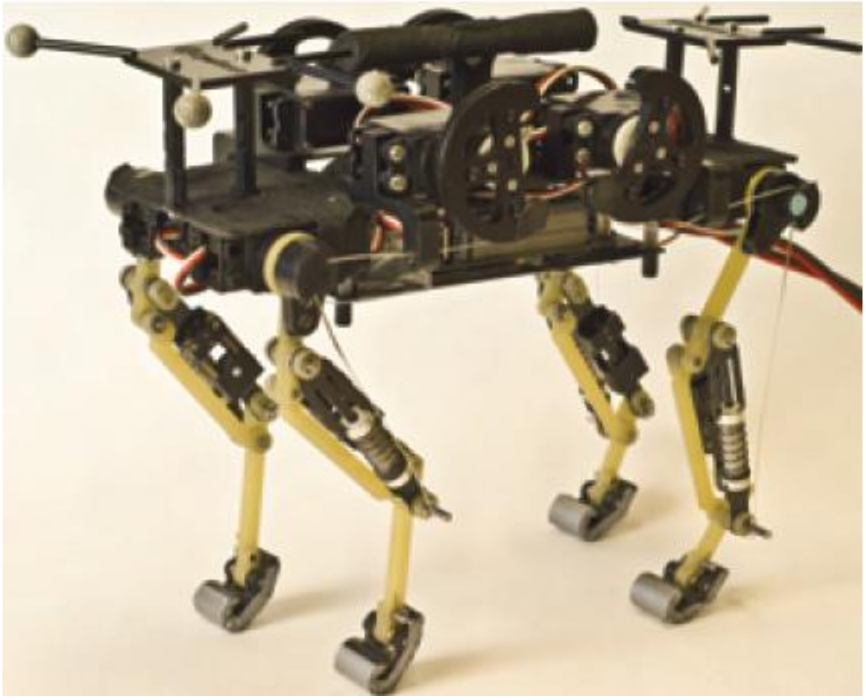


Figure 3. Robot *Cheetah-cub*

From the table we can see that the three robots are of quite different weight and size. Besides, the *BigDog* is designed for large payload carrying capability and high adaptively to the high-level irregularity of the terrain. As a result, for the robot *BigDog*, the most important design factor is the actuation system and the leg structure for supporting large weight and adapt to different terrain. For the robot *Rush*, since it is designed for running, several issues (e.g. the efficient exchange of energy, alleviation of impact damage, load decreases, etc.) have to be taken into account in the mechanical design. The robot *Cheetah-cub* is a small quadruped robot for high efficiency and fast working. As a result, the leg structure and the joint actuation has to be specially designed for this purposed. In the following part, the detailed mechanical design and the actuation design is studied and compared between these three robots.

2.1. Comparison of the mechanical design of the robot legs

The robot leg design for the three robots are shown in Figure 4, 5, 6 respectively. From the figures we can see that the three robots generally shares similar leg configuration with lips, knees and toes. However, due to the different focus of the design, the design of the robot legs vary significantly from each other.

For the robot *BigDog*, since the adaptability to the irregularity of the terrain is of big concern, the robot legs are designed with higher degree of freedom. As we can see from the figure 4 (a), each of the four legs has four degrees of freedom: one passive linear pneumatic compliance in the lower leg, one powered knee joint, and two powered hip joints. The passive linear pneumatic joint in the lower leg main absorbs the impact force and provides damping to the robot. The two powered hip joints provide not only forward and backward movement ability of the leg but also the side way movement ability. This ability greatly increase the adaptively of the *BigDog*. The large degree of freedom of the legs equips the *BigDog* with agile motion ability and also great recovery ability from falling down or slippery.

The heart of *BigDog*'s power is a 17 hp 2 cylinder combustion engine which drives a hydraulic variable displacement pump, providing 3000 PSI system pressure. This high pressure oil is distributed via a central manifold to the twelve custom hydraulic actuators as shown in figure 4 (b). The actuation system for *BigDog* is powerful and stiff enough for support the large weight of *BigDog*.

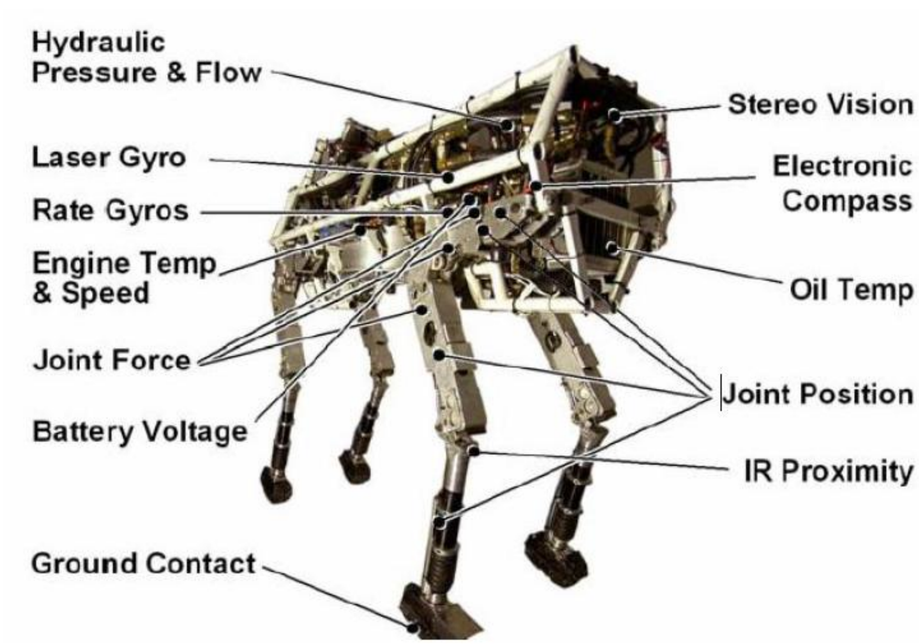


Figure 4 (a). Mechanical design for the robot *BigDog*

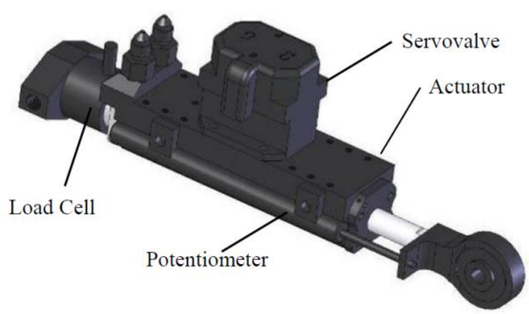


Figure 4 (b). The actuation system of *BigDog*

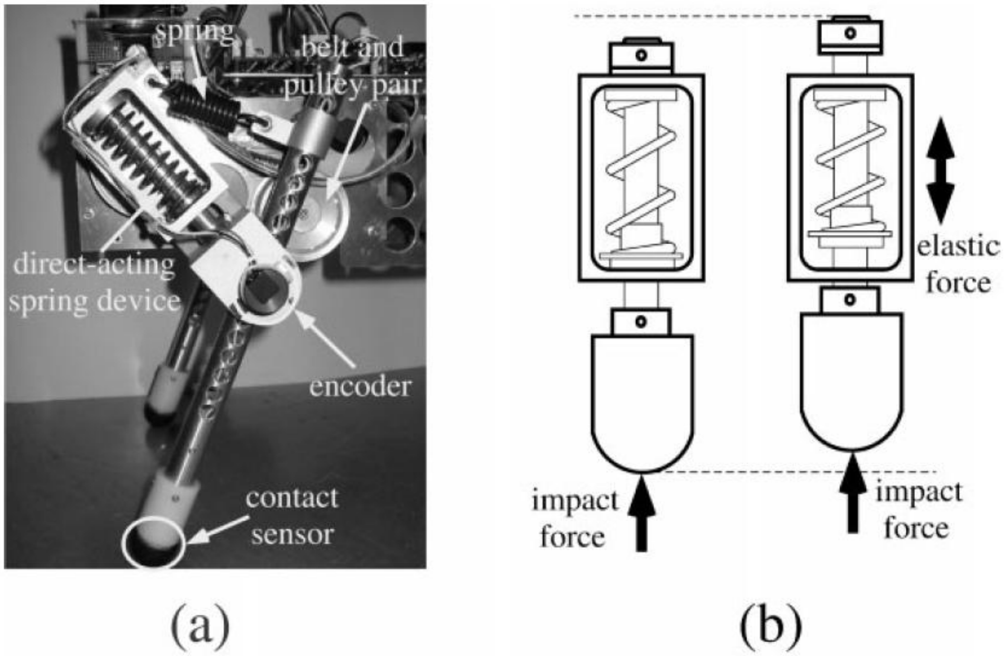


Figure 5. The mechanical design of robot leg for robot *Rush*

For the robot *Rush*, as illustrated in figure 5(a), the leg consists of an upper and a lower part. Both parts are considered to be a chain of two rigid segments with a spring about 20 kN/m, so the knee joint is always passive. The toe is narrow, using a hemispheric hard gum and providing a good approximation to a point of support. In the upper part, a special mechanism, referred to as a direct-acting spring device (see figure 5(b)), is mounted to absorb impact force, so the impact damage acting on the shaft of each joint during rebound with the ground can be alleviated. Each hip is actuated by a d.c. motor of 27.5W power, a three-stage planetary gearbox, and a belt and pulley pair, with about a 19 reduction ratio, providing good amplification of the torque and compliance of the joints. In addition, a belt and pulley pair also isolate impact forces from the motor shaft. The design provide the robot good impact absorption in the running movement and high energy efficiency.

For the robot *Cheetah-Cub*, it adapts the advanced spring-loaded pantograph (ASLP) leg configuration. The ASLP pantograph behavior reflects the l_1–l_3 angular relationship in animals well, but not between midstance and toe-off. During this time, the distal (l3 segment deflects further, due to high external load acting on its elastic elements (Figure 7). For the ASLP leg, a biarticulate spring element was implemented as a replacement of the rigid pantograph l2−prox segment. An in-series spring-loaded foot element replaced the cylinder shaped foot design of the SLP leg. The three-segment, spring-loaded pantograph functionally more similar to the biological counterpart. The hip actuator is mounted directly between the robot’s body and leg. The knee actuator, acting in parallel to the main leg spring, is mounted proximally. The role of the knee actuator is only to retract the leg (by pulling on a cable) and not to extend the leg (leg extension is solely due to the springs).

The robot's legs are each actuated by two RC servo motors, both actuators are mounted proximally. The knee/elbow actuator is attached to the body's center. It actively flexes the leg via a cable mechanism, as an antagonist to the diagonal spring. This cable mechanism acts as an automatic decoupling mechanism, it goes slack if external forces compress the leg. *Cheetah-cub* robot's hip/shoulder actuator is directly mounted between body and leg. It protracts (swings forward) and retracts (swings backward) the front or hind leg. The mechanical structure and the actuation system are specifically designed for this kind of small size and animal-like fast working robots. The servo motors are capable for providing the enough torque and precise position control for this robot. The ASLP leg simulates the walking *cheetah* and the passive knee joint improves the efficiency of the robots.

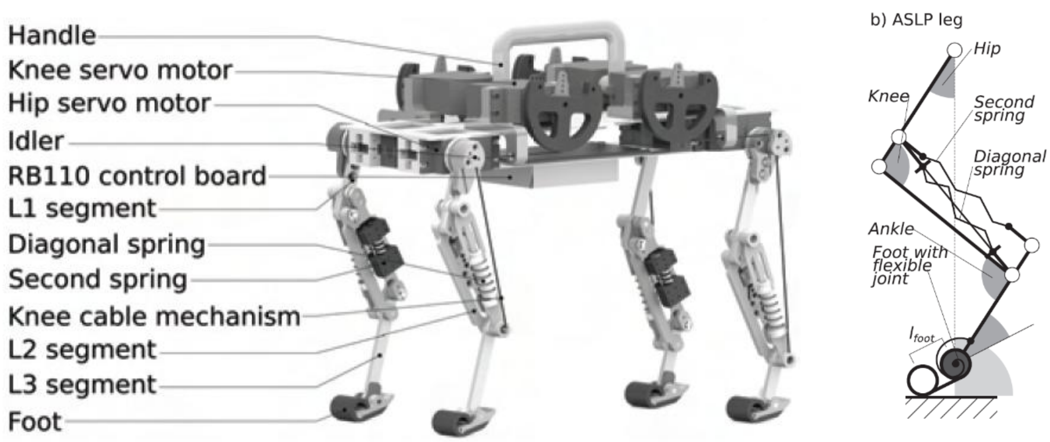


Figure 6. The mechanical design for the robot *Cheetah-Cub*

In comparison, based on the purpose of the robot, there will be different design considerations of the robotic leg. For high mobility and higher terrain adaptation capability, the robotic leg shall have more degree of freedom. For the running robots, compact absorption and energy efficient is most important issue. As a result, the robotic leg shall be simple and designed for more damping effect like the robot *Rush*. For small size of robotic designed for the agility and efficiency, the ASLP leg design which simulates more of the quadruped animals. For the robot *Cheetah-cub*, each of the leg has only two degree of freedom, but the ASLP leg design provide good mobility and agility.

The actuation system design is also closely linked to the weight and focus of the robots. For the *BigDog*, since it is designed to carry large payload and the velocity is not the main concern, the actuation system is made up of the oil pump which is powerful, but not as quick as the motors. For the running robot *Rush*, the DC motor is applied with gear box of reduction ratio 19 to ensure the force generated and the fast reaction. For the small size robots like *Cheetah-cub*, the servo motor is ideal to provide the actuation. The servo motor is of enough torque and fast reaction speed and also precise angle control. This will provide a compact solution of the actuation system and also reduce the usage of the encoders to sense the position of the robotic legs.

In conclusion, in the design of the quadruped robots, the mechanical design shall be closely linked to the focus, or the desired functionality of the robots. Then the structure and actuation system shall be carefully selected so to provide the best solution.

2.2. General control structure of the quadruped robots

The control algorithm is also investigated in the literature review. In particular, the control algorithm of the *BigDog* is presented in this report. The control structure of the *BigDog* is shown in figure 7.

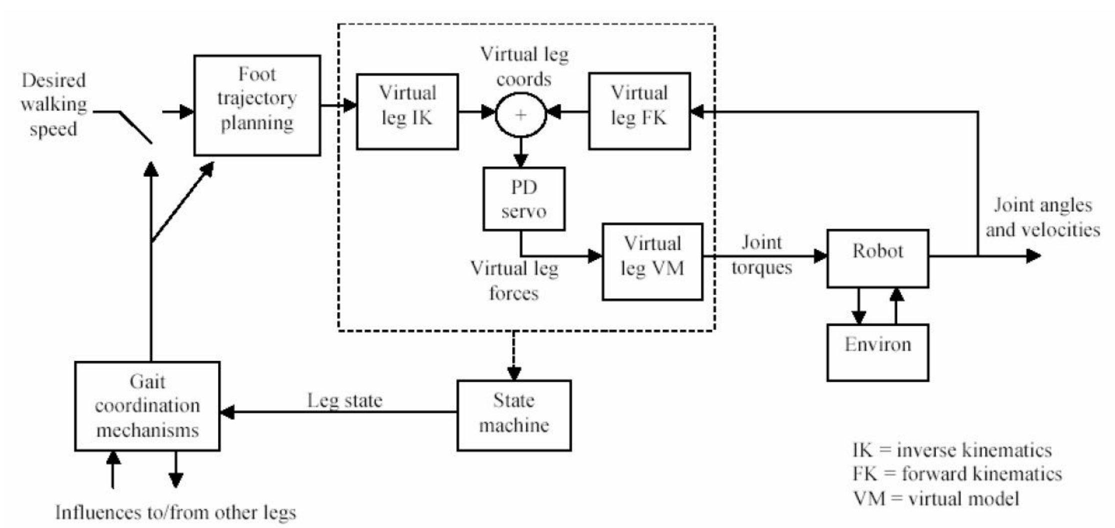


Figure 7 Control structure of the *BigDog*

From figure 7 we can see that the basic walking control uses the control system includes a foot trajectory planning, Gain coordination mechanisms and the leg control algorithm as shown in the dashed box. A gait coordination algorithm, responsible for inter-leg communication initiates leg state transitions to produce different stable quadruped gaits. In the leg control algorithm, a PD servo controller is used with a virtual leg model to coordinate the legs joints and control the legs to follow the desired trajectory.

In this project, to enhance the understanding of the modeling and the control of the quadruped robots, a simplified quadruped robot is proposed and modeled. The detailed modeling of the leg will be introduced in Chapter 3. Then Chapter 4 presents the simulation and the animation of the robot legs. Chapter 5 shows the controller design for the robot leg to follow a desired trajectory using computed torque method and the adaptive control method.

3. Kinetic and dynamic modeling of the quadruped robot leg

In this project, we focus on building a model for a simplified quadruped robot as shown in figure 8 (a). Each leg is considered to be identical with two links and each link is assumed to be cylinders with zero radiuses and the mass of each link is assumed to be lumped in the middle of the link. For each leg, assume that I_i is the moment of inertia about an axis coming out of the page passing through the center of mass, m_i, of link i; l_i and l_{ic} are the length of link, and the distance from rotating joint to the center of mass of link as indicated; and $\tau\ i$ is the driving torque at the joint i (i=1, 2). Parameters of the each leg are given in the table below.

Parameter	Description	Value
m_1	Mass of link 1	$1.151kg$
m_2	Mass of link 2	$2.142kg$
l_1	Length of link 1	$0.31m$
l_1	Length of link 2	$0.34m$
I_1	Inertia of link 1	$0.3kgm^2$
I_2	Inertia of link 2	$0.3kgm^2$

Table 1: Parameters of the robotic dog

To build the model of the robotic leg, the leg parameter definition is shown in the figure 8 (b), let m_i and l_i be the mass and length of link i, l_{ci} is the distance from joint i-1 to the center of mass of the link i, and I_i be the moment of inertia link i about axis coming out of the page passing through the center of mass of link i.

Figure 8(a). Simplied model for a quadruped robot

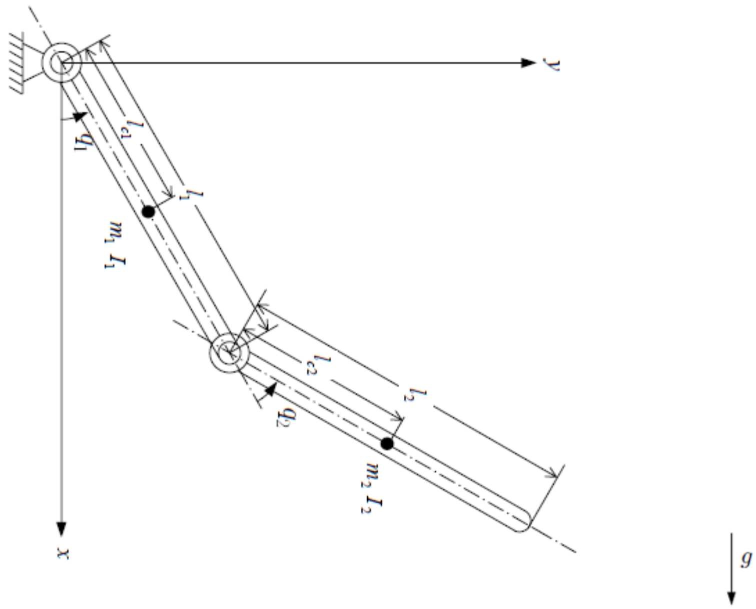


Figure 8 Coordinate system of 2-dof robotic leg

3.1 Kinetic modeling

The position of the leg tip is given as:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} l_1 \cos q_1 + l_2 \cos(q_1+q_2) \\ l_1 \sin q_1 + l_2 \sin(q_1+q_2) \end{bmatrix}$$

Differential form:

$$\mathrm{dx} = \frac{\partial \mathrm{x}(q_1, q_2)}{\partial q_1} dq_1 + \frac{\partial \mathrm{x}(q_1, q_2)}{\partial q_2} dq_2$$

$$\mathrm{dy} = \frac{\partial \mathrm{y}(q_1, q_2)}{\partial q_1} dq_1 + \frac{\partial \mathrm{y}(q_1, q_2)}{\partial q_2} dq_2$$

let $\mathrm{dX} = \mathrm{Jdq}$

where $\mathrm{dX} = \begin{bmatrix} \mathrm{dx} \\ \mathrm{dy} \end{bmatrix}, \mathrm{dq} = \begin{bmatrix} \mathrm{d}q_1 \\ \mathrm{d}q_2 \end{bmatrix}$ and J is the Jacobian matrix

$$\mathrm{J} = \begin{bmatrix} -l_1 \sin q_1 - l_2 \sin(q_1+q_2) & -l_2 \sin(q_1+q_2) \\ l_1 \cos q_1 + l_2 \cos(q_1+q_2) & l_2 \cos(q_1+q_2) \end{bmatrix}$$

This can also be written as

$$\mathrm{V} = \mathrm{J}\dot{q}$$

Where V denotes the dog leg tip velocity vector and $\dot{q}$ denotes the leg joint velocity.

3.2 Dynamic modeling

The position of $\mathrm{m}_i (i = 1{,}2)$ are given by

$$\begin{bmatrix} x_{c1} \\ y_{c1} \end{bmatrix} = \begin{bmatrix} l_{c1}\cos q_1 \\ l_{c1}\sin q_1 \end{bmatrix}$$

$$\begin{bmatrix} x_{c2} \\ y_{c2} \end{bmatrix} = \begin{bmatrix} l_1\cos q_1 + l_{c2}\cos(q_1+q_2) \\ l_1\sin q_1 + l_{c2}\sin(q_1+q_2) \end{bmatrix}$$

So the end point velocity can be obtained as:

$$v_{c1} = \begin{bmatrix} v_{c1x} \\ v_{c1y} \end{bmatrix} = \begin{bmatrix} -l_{c1}\dot{q}_1\sin q_1 \\ l_{c1}\dot{q}_1\cos q_1 \end{bmatrix}$$

$$v_{c2} = \begin{bmatrix} v_{c2x} \\ v_{c2y} \end{bmatrix} = \begin{bmatrix} -l_1\dot{q}_1\sin q_1 - l_{c2}(\dot{q}_1+\dot{q}_2)\sin(q_1+q_2) \\ l_1\dot{q}_1\cos q_1 + l_{c2}(\dot{q}_1+\dot{q}_2)\cos(q_1+q_2) \end{bmatrix}$$

The total kinetic energy is:

$$\mathrm{K} = \frac{1}{2}\mathrm{m}_1 v_{c1}{}^T v_{c1} + \frac{1}{2}I_1\dot{q}_1{}^2 + \frac{1}{2}\mathrm{m}_2 v_{c2}{}^T v_{c2} + \frac{1}{2}I_2(\dot{q}_1+\dot{q}_2)^2$$

$$= \frac{1}{2}\mathrm{m}_1 l_{c1}{}^2\dot{q}_1{}^2 + \frac{1}{2}I_1\dot{q}_1{}^2 + \frac{1}{2}\mathrm{m}_2 l_1{}^2\dot{q}_1{}^2 + \mathrm{m}_2 l_1 l_{c2}\dot{q}_1(\dot{q}_1+\dot{q}_2)\cos(q_2)$$
$$+ \frac{1}{2}\mathrm{m}_2 l_{c2}{}^2(\dot{q}_1+\dot{q}_2)^2 + \frac{1}{2}I_2(\dot{q}_1+\dot{q}_2)^2$$

The potential energy of the manipulator is

$$\mathrm{P} = \mathrm{m}_1 g y_{c1} + \mathrm{m}_2 g y_{c2}$$

$$= \mathrm{m}_1 g l_{c1}\sin q_1 + \mathrm{m}_2 g(l_1\sin q_1 + l_{c2}\sin(q_1+q_2))$$

Use Lagrange equation method:

$$\mathrm{L} = \mathrm{K} - \mathrm{P}$$

$$\frac{d}{d_t}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau$$

Where $\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$ is the generalized torque vector.

$$\frac{\partial L}{\partial \dot{q}_1} = \left(\mathrm{m}_1 l_{c1}{}^2 + I_1 + \mathrm{m}_2 l_1{}^2 + \mathrm{m}_2 l_{c2}{}^2 + I_2\right)\dot{q}_1 + \left(\mathrm{m}_2 l_{c2}{}^2 + I_2\right)\dot{q}_2$$
$$+ 2\mathrm{m}_2 l_1 l_{c2}\cos\left(q_2\right)(\dot{q}_1+\dot{q}_2)$$

$$\frac{d}{d_t}\frac{\partial L}{\partial \dot{q}_1} = \left(\mathrm{m}_1 l_{c1}{}^2 + I_1 + \mathrm{m}_2 l_1{}^2 + \mathrm{m}_2 l_{c2}{}^2 + I_2\right)\ddot{q}_1 + \left(\mathrm{m}_2 l_{c2}{}^2 + I_2\right)\ddot{q}_2$$
$$+ 2\mathrm{m}_2 l_1 l_{c2}\cos\left(q_2\right)(\ddot{q}_1+\ddot{q}_2) - 2\mathrm{m}_2 l_1 l_{c2} sin\left(q_2\right)(\dot{q}_1\dot{q}_2 + \dot{q}_2{}^2)$$

$$\frac{\partial L}{\partial q_1} = -(\mathrm{m}_1 g l_{c1} + \mathrm{m}_2 g l_1)\cos q_1 - \mathrm{m}_2 g l_{c2} cos(q_1+q_2)$$

$$\frac{\partial L}{\partial \dot{q}_2} = \mathrm{m}_2 l_1 l_{c2}\cos\left(q_2\right)\dot{q}_1 + \left(\mathrm{m}_2 l_{c2}{}^2 + I_2\right)(\dot{q}_1+\dot{q}_2)$$

$$\frac{d}{d_t}\frac{\partial L}{\partial \dot{q}_2} = -\mathrm{m}_2 l_1 l_{c2}\sin\left(q_2\right)\dot{q}_1\dot{q}_2 + \mathrm{m}_2 l_1 l_{c2}\cos\left(q_2\right)\ddot{q}_1 + \left(\mathrm{m}_2 l_{c2}{}^2 + I_2\right)(\ddot{q}_1+\ddot{q}_2)$$

$$\frac{\partial L}{\partial q_2} = -\mathrm{m}_2 l_1 l_{c2} \dot{q}_1 (\dot{q}_1 + \dot{q}_2) \sin(q_2) - \mathrm{m}_2 g l_{c2} cos(q_1 + q_2)$$

For the above kinetic energy can be further rewritten as:

$$\mathrm{K} = \frac{1}{2} \dot{q}^T D(q) \dot{q}$$

Where $D(q)$ is the inertia matrix given by

$$D(q) = \begin{bmatrix} p_1 + p_2 + 2p_3 \cos q_2 & p_2 + p_3 \cos q_2 \\ p_2 + p_3 \cos q_2 & p_2 \end{bmatrix}$$

With

$$\begin{cases} p_1 = \mathrm{m}_1 {l_{c1}}^2 + \mathrm{m}_2 {l_1}^2 + I_1 \\ p_2 = \mathrm{m}_2 {l_{c2}}^2 + I_2 \\ p_3 = \mathrm{m}_2 l_1 l_{c2} \end{cases}$$

Thus we can have

$$\mathrm{c}_{111} = \frac{1}{2} \frac{\partial d_{11}}{\partial q_1} = 0$$

$$\mathrm{c}_{121} = \mathrm{c}_{211} = \frac{1}{2} \frac{\partial d_{11}}{\partial q_2} = -p_3 \sin q_2$$

$$\mathrm{c}_{221} = \frac{\partial d_{12}}{\partial q_2} - \frac{1}{2} \frac{\partial d_{22}}{\partial q_1} = -p_3 \sin q_2$$

$$\mathrm{c}_{112} = \frac{\partial d_{21}}{\partial q_1} - \frac{1}{2} \frac{\partial d_{11}}{\partial q_2} = p_3 \sin q_2$$

$$\mathrm{c}_{122} = \mathrm{c}_{212} = \frac{1}{2} \frac{\partial d_{22}}{\partial q_1} = 0$$

$$\mathrm{c}_{222} = \frac{1}{2} \frac{\partial d_{22}}{\partial q_2} = 0$$

Using these, the Coriolis and centrifugal matrix $\mathrm{C}(\mathrm{q}, \dot{\mathrm{q}})$ can be obtained as

$$\mathrm{C}(\mathrm{q}, \dot{\mathrm{q}}) = \begin{bmatrix} -p_3 \dot{q}_2 \sin q_2 & -p_3 (\dot{q}_1 + \dot{q}_2) \sin q_2 \\ p_3 \dot{q}_1 \sin q_2 & 0 \end{bmatrix}$$

The potential energy of the manipulator is

$$\mathrm{P} = \mathrm{m}_1 g y_{c1} + \mathrm{m}_2 g y_{c2}$$

$$= \mathrm{m}_1 g l_{c1} \sin q_1 + \mathrm{m}_2 g l_1 \sin q_1 + l_{c2} \sin(q_1 + q_2)$$

Which leads to the gravitation force vector $\mathrm{G}(\mathrm{q})$ is given by

$$G(q) = \frac{\partial P}{\partial q} = \begin{bmatrix} p_4 g cos q_1 + p_5 g cos(q_1 + q_2) \\ p_5 g cos(q_1 + q_2) \end{bmatrix}$$

Where

$$\begin{cases} p_4 = m_1 l_{c2} + m_2 l_1 \\ p_5 = m_2 l_{c2} \end{cases}$$

Thus the dynamic model of the dog leg is given as

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau$$

where $\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$ is the generalized torque vector.

4. Simulation and the animation of the quadruped robot

4.1. Software Introduction

Automated Dynamic Analysis of Mechanical Systems (ADAMS) is a multibody dynamics simulation software equipped with Fortran and C++ numerical solvers. ADAMS was originally developed by Mechanical Dynamics Incorporation which then was acquired by MSC Software Corporation. Adams has been proved as very essential to VPD (Virtual Prototype Development) through reducing product time to market and product development costs. MSC Software generally supports the two operating systems Linux and Microsoft Windows.

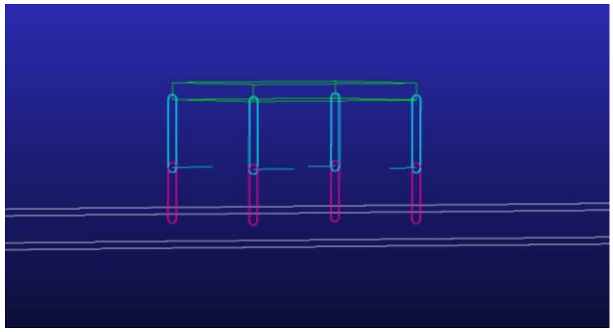
As the world's most famous and widely used Multibody Dynamics (MBD) software, Adams improves engineering efficiency and reduces product development costs by enabling early system-level design validation. Engineers can evaluate and manage the complex interactions between disciplines including motion, structures, actuation, and controls to better optimize product designs for performance, safety, and comfort. Along with extensive analysis capabilities, Adams is optimized for large-scale problems, taking advantage of high performance computing environments.

Utilizing multibody dynamics solution technology, Adams runs nonlinear dynamics in a fraction of the time required by FEA solutions. Loads and forces computed by Adams simulations improve the accuracy of FEA by providing better assessment of how they vary throughout a full range of motion and operating environments.

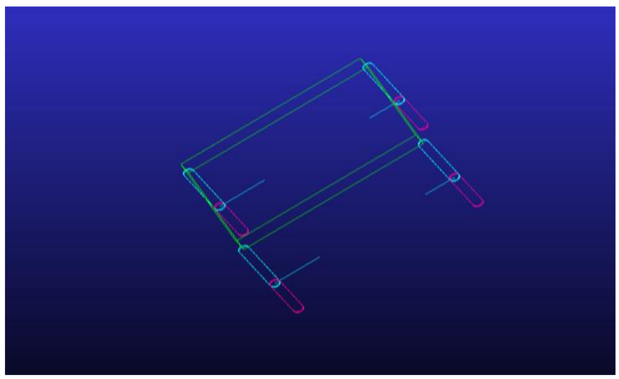
4.2 Model design

Adams/View software is employed to draw the prototype of the robotic dog. In order to simplify the work, the following assumptions are made: (1) The torque provided by the motor at the joints can be arbitrary value. (2) All the legs and the body are rigid body and the fiction at the connection points are neglected. (3) The floor which the robot walks on is flat. (4) Only the joints at the ankle and the hip are taken into consideration, namely, a 2-DOF robotic arm is used to simulate the legs.

The prototype of the dog is shown as follows.



(a)



(b)

Figure 9 Drawing of robot dog under Adams

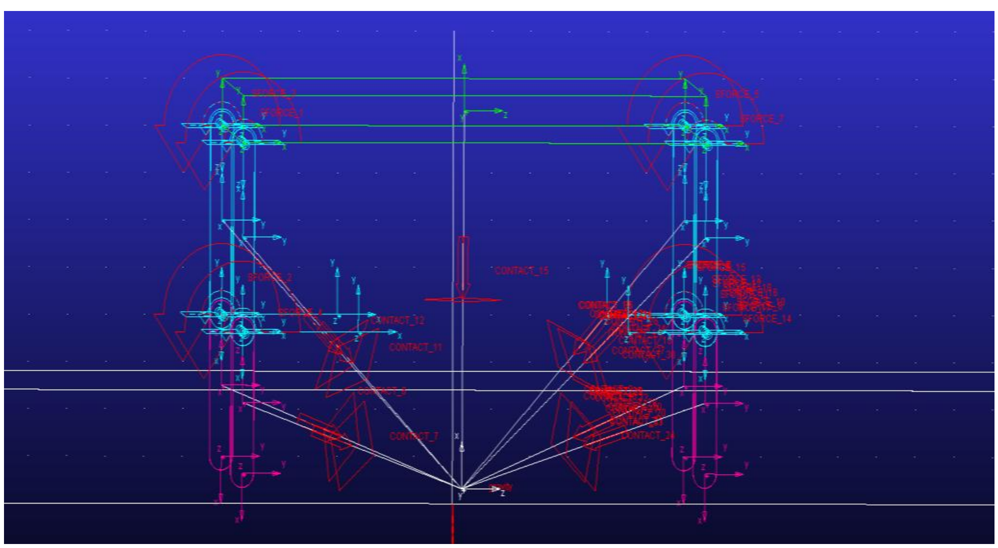


Figure 10 Dog model with joint torque and contact force

To achieve co-simulation work with Matlab, the prototype is exported as an "adams_sys.mdl" file as follows. Based on this file, a composite controller is developed using Matlab\Simulink.

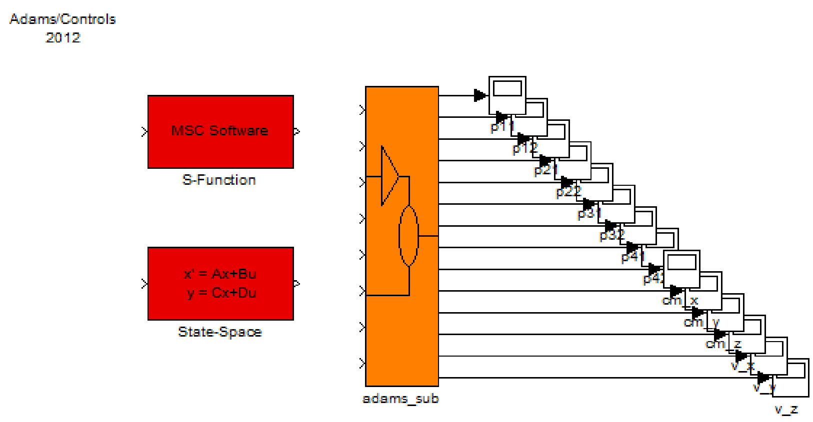


Figure 11 adams_sys.mdl file

4.3 Control System Design

The control system is composed of leg motion controllers and a pattern controller as shown below. The leg motion controllers drive all the joint actuators of the legs so as to realize the desired motions that are generated by the gait pattern controller. The gait pattern controller involves nonlinear oscillators corresponding to each leg. The gait pattern controller receives the commanded signal of the nominal gait pattern as the reference.

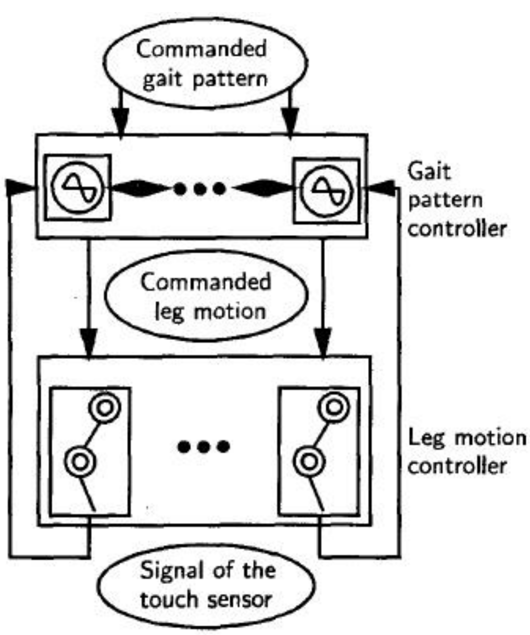


Figure 12 Structure of control system

4.4. Gait Design

For the animation part, the trotting gait is studied. This gait requires the diagonal legs move as a virtual leg [4]. As the following figure shows, where the line means the supporting phase and the blank means the swinging phase.

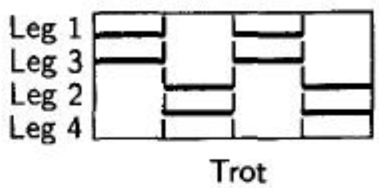


Figure 13 Phases of trotting gait

The ratio of the support stage to the whole period is called "duty factor". In this project, the duty factor is chosen as 0.5. As the following figure shows, *z* is defined as the displacement from the first joint to the ground and it is generated as sine function. And *x* is designed to track cosine trajectory. The amplitude of the oscillation in x-axis is 0.1; the amplitude on the z-axis is 0.05. $m = \sqrt{x^2 + y^2}$ and $\theta_1 = 180^{\circ} - u_1$, $\theta_2 = 180^{\circ} - u_2$, where these angles are shown in the following figure.

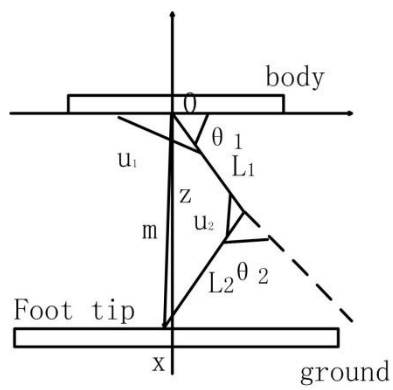


Figure 14 Coordinate of single leg

The pattern generator is designed as follows.

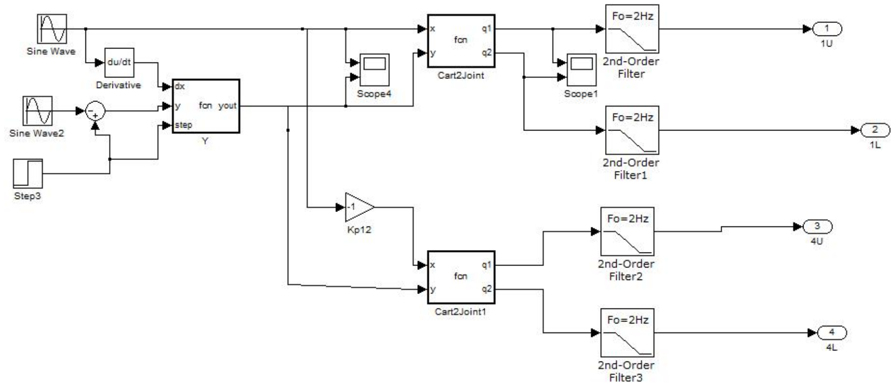


Figure 15 Structure of pattern generator

The above system generates the reference signal for the first virtual leg. The "Cart2joint" module provides the transform of the trajectory from Cartesian space to the joint space. And for the first virtual leg, four joints are calculated in total representing the upper joint and the lower joint of the first two diagonal joints. A delay module (following figure) is added to provide the reference signal of another pair of diagonal legs.

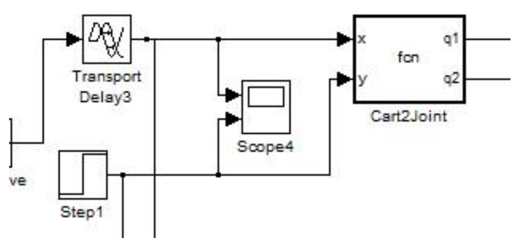


Figure 16 Delay module for leg 2 and leg 3

To guarantee the joint value to track the given reference pattern, the leg motion controller is designed based on PD controller. The motion controller is shown below. Due to the large size of the plot and the similarity of all the control module, only the controllers for the first two joints are given.

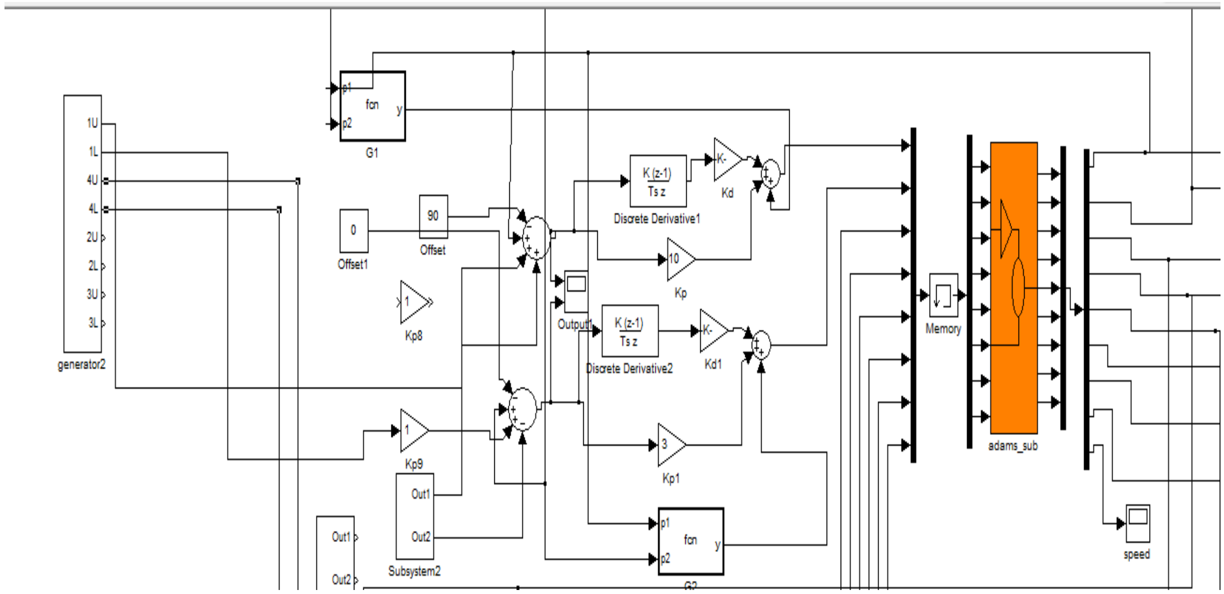


Figure 17 Leg motion controller

4.5. Simulation

All the parameters of K_p are chosen to be 0.021 and the K_d are selected to be 10 for the upper joints and 3 for the lower joints. The following results show four typical phases of trotting. From the result, we can see that the left fore-leg and right rear-leg take the same gaits from the supporting phase to swing phase and supporting phase again, which is complete trotting circle.

Figure 18 Trotting gait of the simulated dog using Matlab-Adams Co-simulation

5. Control of the quadruped robot

5.1. Computed torque method controller design

In the previous chapter, the modeling of the quadruped robot leg and the PD control is simulated and the animation is also presented. In this part, a computed torque method controller based controller is designed for a better control action of the robot leg for trajectory following.

The computed torqued method is quite straight forward. The idea is to convert the nonlinear dynamic of the robotic legs into a simple double integrator system with the help of the existing knowledge of the model. For instance, the typical model of the robotic leg is:

$$\mathrm{M(q)}\ddot{q} + C(q,\dot{q})\dot{q} + G = \tau$$

Let

$$\tau = M(q)u + C(q,\dot{q})\dot{q} + G \tag{1}$$

then the original equation becomes

$$\mathrm{M(q)}\ddot{q} + C(q,\dot{q})\dot{q} + G = M(q)u + C(q,\dot{q})\dot{q} + G$$

So we can obtained the simple double integrator system dynamics

$$\ddot{q} = u$$

As a result, we can use any controller to design for this system to obtain the u value, then the required torque for the joints could be obtained through Eqn (1). In this project, a linear quadratic regulator controller (LQR) is designed for this double integrator system since the LQR controller has an infinite gain margin and at least 60^0 of phase margin. Written in state space form with state variable $\mathrm{x} = [\mathrm{q}\ \dot{\mathrm{q}}]^T$, the double integrator system could be denoted as:

$$\dot{x} = Ax + bu$$

where $\mathrm{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $\mathrm{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. The cost function for a LQR controller is defined as

$$\mathrm{J} = \int_0^\infty (x^T Q x + u^T R u) dt$$

where Q and R are symmetrical positive definite matrix. The feedback control law that minimized the value of the cost is $u = -Kx$ where K is given by $K = R^{-1} B^T P$ and P is the solution of the Riccati equation:

$$\mathrm{A}^\mathrm{T} P + PA - PBR^{-1} B^T P + Q = 0$$

In this project, choose $\mathrm{Q} = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$ and R = 1 so to regulate the states more, we can get the controller gain K = $[2 \ 2.8284]^\mathrm{T}$. In this project, assume that the angular position and the velocity of each joint is measurable, then the state feedback controller could be applied.

The simulation of the robotic leg is done through the method of direct computation, where

$$\ddot{q} = M(q)^{-1}(\tau - C(q, \dot{q})\dot{q} - G - J^T f)$$

$$\dot{\mathrm{q}}_{new} = \dot{q} + \ddot{q}\Delta t$$

$$q_{new} = \mathrm{q} + \dot{q}\Delta t + \frac{1}{2}\ddot{q}\Delta t^2$$

Since typically the force from the ground is hard to measure, in the controller design, the force from the ground is not taken into consideration and may be treated as disturbance of the system. In the simulation, the force on the ground is simulated as $\mathrm{f} = \sin(\mathrm{t})$.

The system reference input is $\mathrm{q} = \begin{bmatrix} \sin(0.5t) \\ \sin(0.2t) \end{bmatrix}$ and the simulation result for the computed torque method with LQR controller is shown as figure 19 and figure 20. The red plot is the reference plot and the blue line is the output of the angle and angular rate.

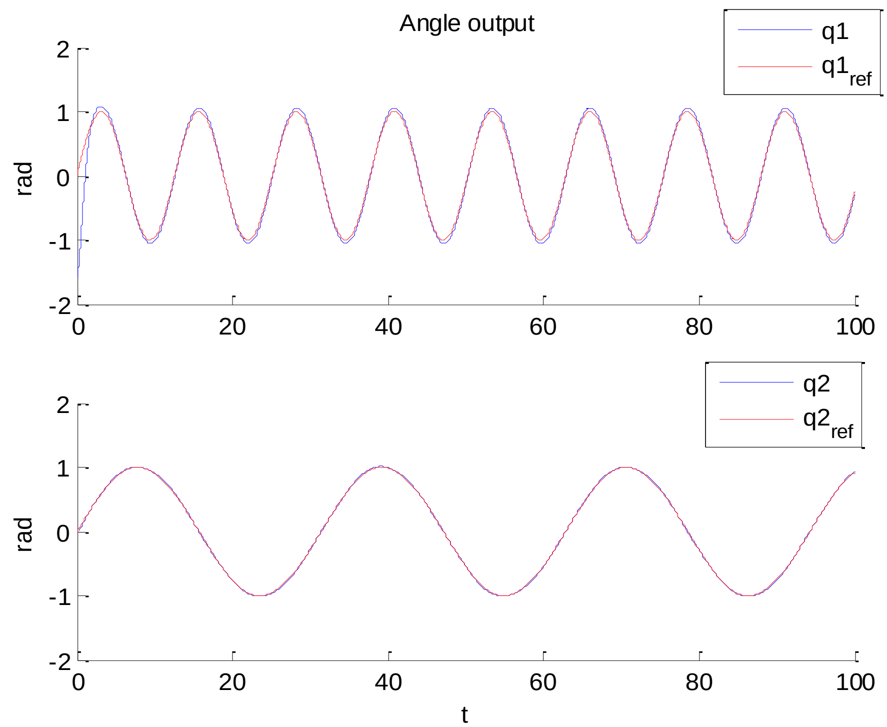


Figure 19. The angle output of the computed torque method

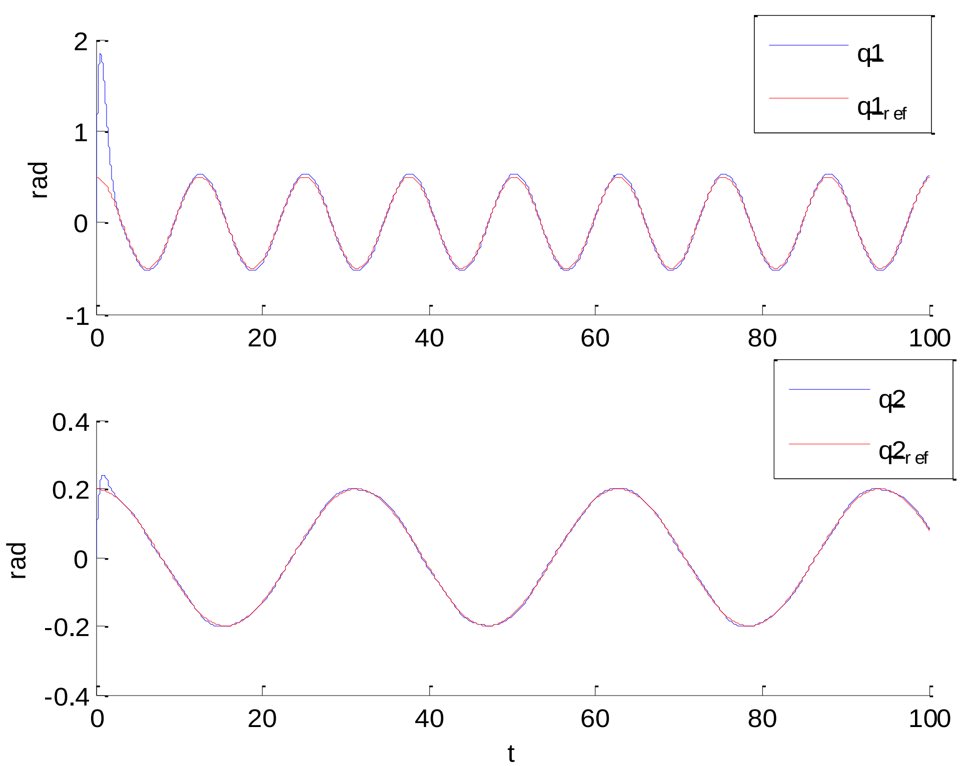


Figure 20. The angular rate output of computed torque method

From the figure 19 and figure 20 we can see that even with unknown disturbance (the ground force in this project), the computed torque method with LQR design can still achieve very good tracking result of the desired trajectory.

Next, a simulation work is done with measurement noise of power 0.01, the simulation result is shown as figure 10 and figure 11. In real life, there will be always measurement noise existing however good the sensor is. However, since the computed torque method requires to know the exact information of the model and the measurement, the adaption ability of the computed torque method is tested with measurement noise. From figure 21 and 22 we can see that although the noise presents, the controlled output can still follow the trajectory quite well. This shows the robustness of the computed torque method to measurement noise.

Lastly, the situation where the system parameters are not exact known is simulated. From Eqn. (1), we know that the final torque applied to the system depends on the M(q) and C(q) which may not been exactly known in real life. In this simulation, when compute M(q) and C(q) for the torque, assume that the m_1 = 1 kg and l_1 = 0.2 m which slightly varies from the exact value. The simulation result is shown as figure 23 and figure 24. From the figures we can see that when there are uncertainties presenting in the model parameters, the computed torque method does not perform well. This is the shortage of the computed torque method.

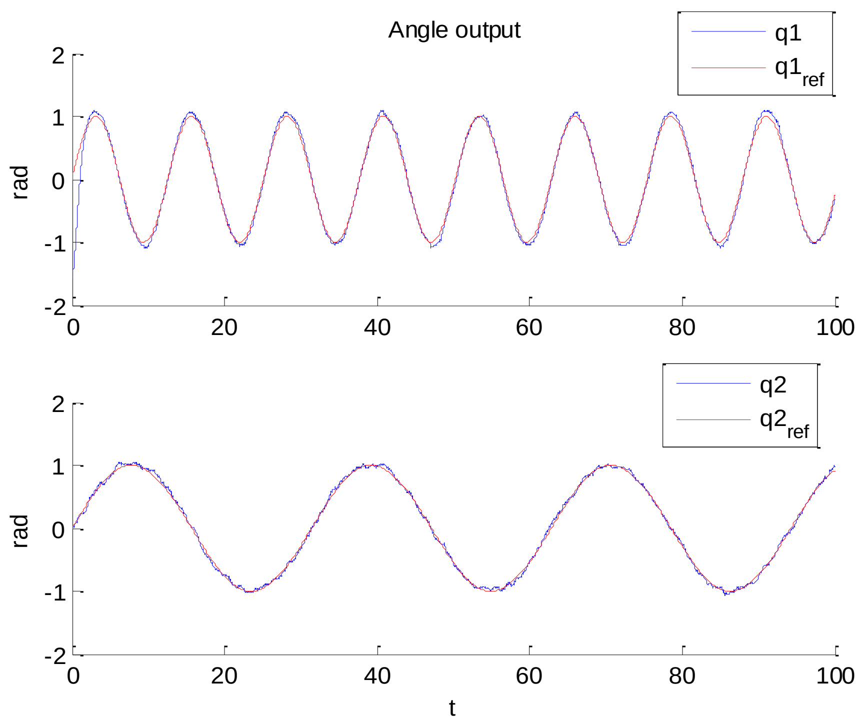


Figure 21. The angle output of computed torque method with measurement noise

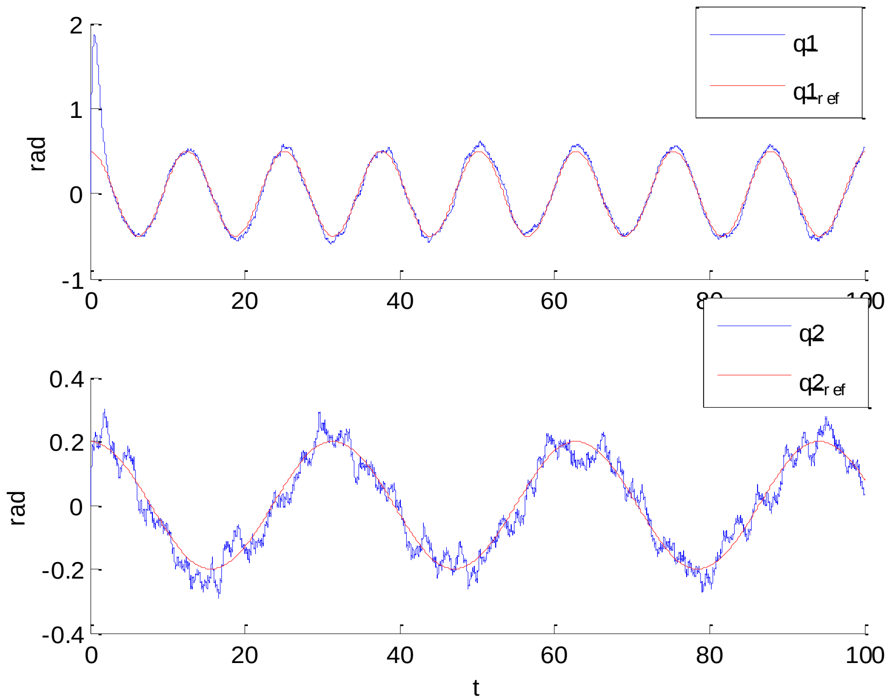


Figure 22. The angular rate output of computed torque method with measurement noise

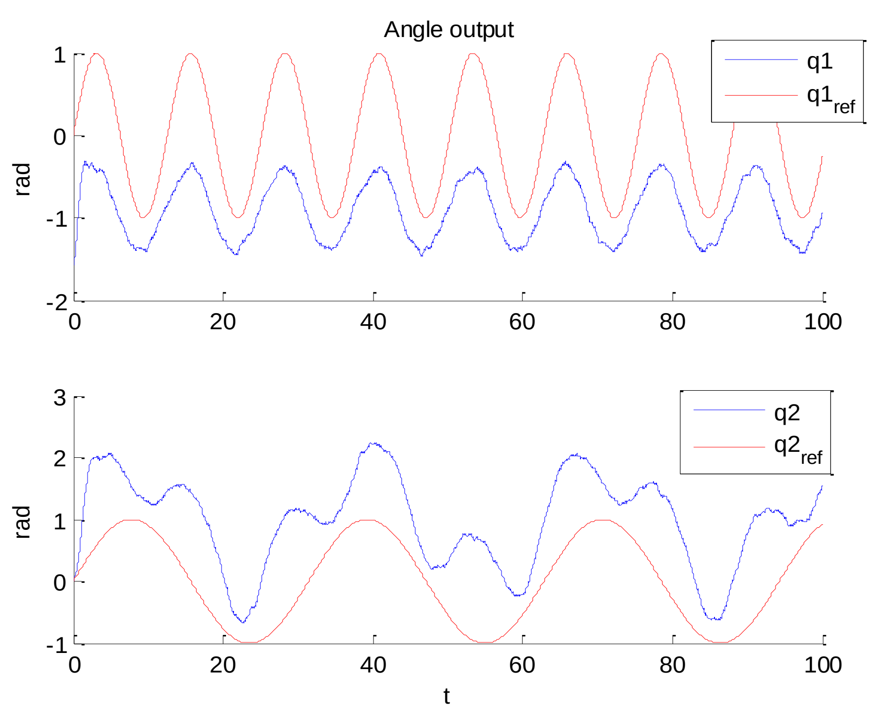


Figure 23. The angle output of computed torque method with parameter uncertainty

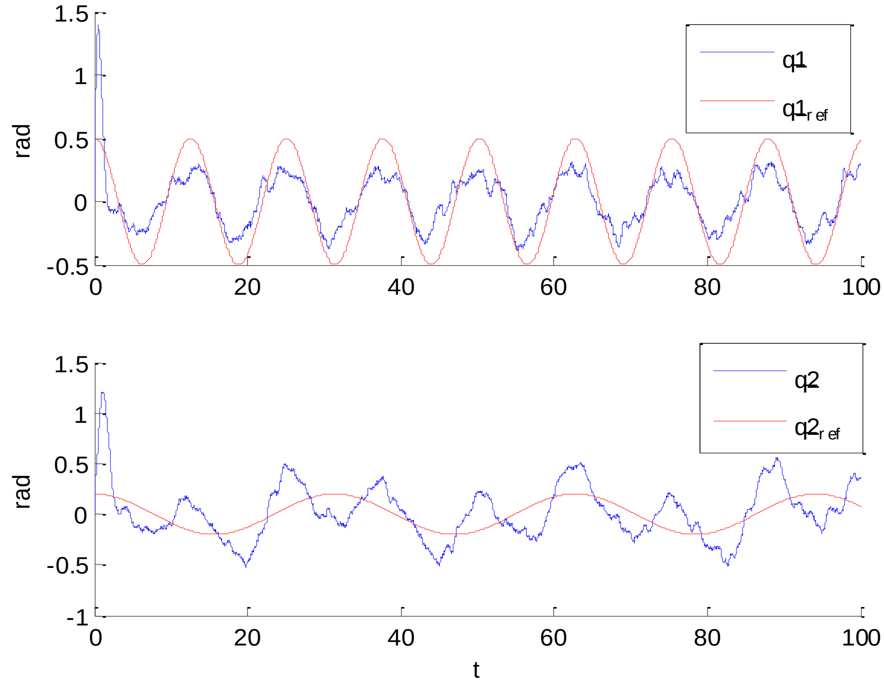


Figure 24. The angular rate output of computed torque method with parameter uncertainty

5.2.The adaptive control of the robotic leg

Since the computed torque method requires the full knowledge of the system, the application is limited. In this section, the adaptive controller is investigated on this robotic leg system which does not require the knowledge of model of the system. The procedure for the adaptive control law is denoted as follows:

1. Find $q_d, \dot{q}_{d,} \ddot{q}_d$, or the position, velocity and acceleration of the desired trajectory;
2. Find the tracking error $e = q_d - q$
3. Define the auxiliary error $\mathrm{r} = \dot{e} + \lambda\mathrm{e}, \lambda > 0$, then if r goes to 0, e and $\dot{e}$ goes to 0.
4. Define $\dot{\mathrm{q}}_r = \dot{\mathrm{q}}_d + \lambda\mathrm{e}$
5. Define $\ddot{q}_r = \ddot{\mathrm{q}}_\mathrm{d} + \lambda\dot{e}$

Find the linear-in-the-parameter dynamics

$$\mathrm{D(q)a} + \mathrm{C(q}, \dot{q})v + G(q) = \Psi(q, \dot{q}, v, a)P$$

where the Ψ is the known regressor of the system and P is the true robot parameter vector which is unknown usually. In the project, the

$\Psi(q,\dot{q},v,a) =$
$$\begin{bmatrix} a_1 & a_1+a_2 & 2a_1c_1+c_2a_2-v_1q_2s_2-(v_1+v_2)s_2v_2 & a_1 & a_1+a_2 & c_1 & c_1 & c_{12} \\ 0 & a_1+a_2 & c_2a_1+s_2q_1v_1 & 0 & a_1+a_2 & 0 & 0 & c_{12} \end{bmatrix}$$
and the parameter P is

$$\mathrm{P} = [m_1lc_1^2 + m_2l_1^2 \quad m_2lc_2^2 \;\; m_2l_1lc_2 \;\; I_1 \;\; I_2 \quad m_1lc_2g \;\; m_2l_1g \;\; m_2lc_2g]^T$$

Then the adaptive control law is given as

$$\tau = \Psi(\mathrm{q},\dot{q},\mathrm{v},\mathrm{a})\hat{P} + Kr$$

where $\hat{P}$ is the estimated parameter vector an the adaptation algorithm is

$$\dot{\hat{P}} = \Gamma^{-1}\Psi^T(q,\dot{q},\dot{q}_r,\ddot{q}_r)r$$

where $\Gamma = \Gamma^{\mathrm{T}} > 0$ is the weighing matrix for the adaptation. It can be proven that as time goes to infinity, $\hat{P}$ is approaching the exact value of P. In this part, simulation is done for this adaptive controller with $\mathrm{K} = 0.9, \lambda = 0.005, \Gamma = 1000 \times \mathrm{I}$, the reference trajectory is

$$\mathrm{qd} = \begin{bmatrix} \sin(2t) + \sin(t) \\ \sin(0.5t) \end{bmatrix}$$

The output of the simulation is shown as figure 25 and 26. From the figures we can see that the controller does not perform well in the beginning 10 s. After 10s of adaptation, the controller performs quite well for following the fast changing reference trajectory. The estimated

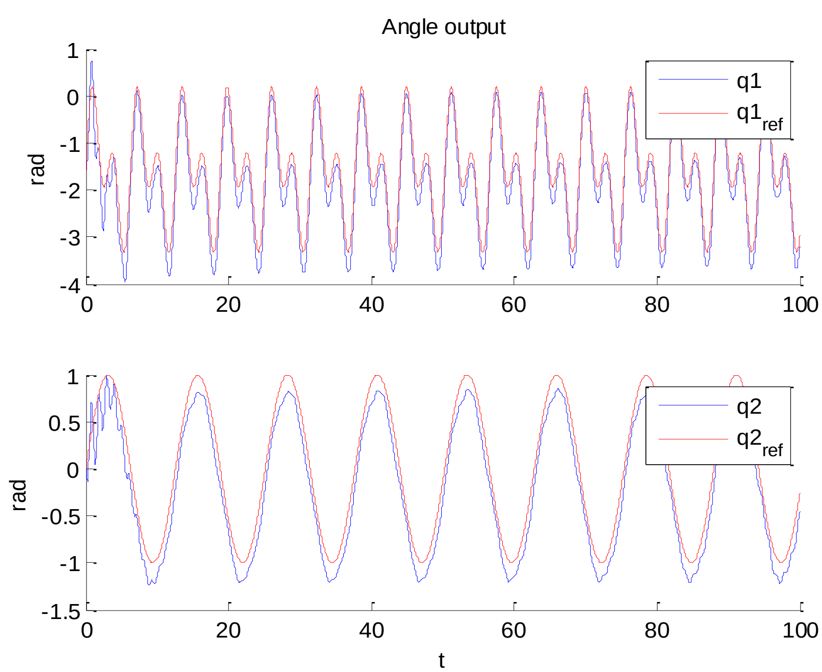


Figure 25. The angle output of the adaptive controller

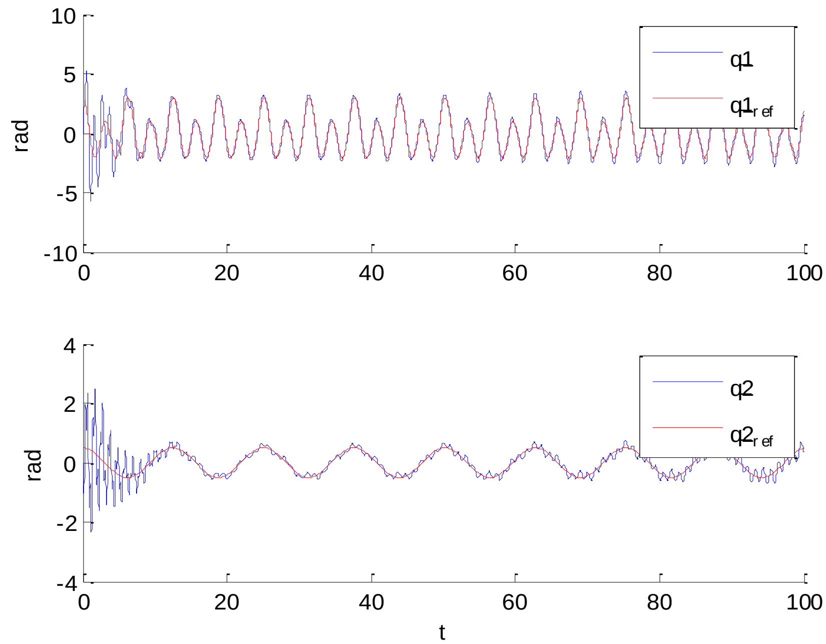


Figure 26. The angular rate output of the adaptive controller

$$\hat{P} = [0.2326\ 0.0608\ 0.1136\ 0.2991\ 0.2989\ 1.9183\ 6.5082\ 3.5697]^T$$

While the exact

$$\mathrm{P} = [0.2325\ 0.0619\ 0.1129\ 0.3000\ 0.3000\ 1.9176\ 6.5074\ 3.5686]^T$$

The estimated value is quite close to the exact value. This shows the great adaptive ability of the adaptive controller. Without knowing the exact parameters of the plant, the adaptive controller has much wider application than the computed torque method.

6. Conclusion

In conclusion, in this project, the literature review has been done for the quadruped robots. The mechanical design and the actuation system design are carefully investigated and compared. The controller structure of the robot *BigDog* is also studied. Then a kinematic and dynamic modeling has been done for a simplified model of the robotic dog leg. Simulation and animation has been done with the aid of PD controller for a better presentation of the modeling. Then based on the modeling, the computed torque method and the adaptive controller are designed and implemented for the robotic dog leg model. Their results are carefully studied and compared. When the system parameters are all known, the computed torque method could give a very good control performance with high tolerance to noise. However, when there are uncertainties present in the model parameters, the computed torque method does not perform well at all. The adaptive controller, in contrast, do not require the system parameters at all. The adaptation process could estimate the system parameters of the LIP form and has wider application in the robotic control.

References

[1] Ge, Shuzhi S., C. C. Hang, and L. C. Woon. "Adaptive neural network control of robot manipulators in task space." Industrial Electronics, IEEE Transactions on 44.6 (1997): 746-752.

[2] Raibert, Marc H., and Ivan E. Sutherland. "Machines that walk." Scientific American 248 (1983): 44-53.

[3] R. Playter ; M. Buehler and M. Raibert, "BigDog", Proc. SPIE 6230, Unmanned Systems Technology VIII, 62302O (May 09, 2006); doi:10.1117/12.684087;http://dx.doi.org.libproxy1.nus.edu.sg/10.1117/12.684087

[4] Spröwitz, A., Tuleu, A., Vespignani, M., Ajallooeian, M., Badri, E., & Ijspeert, A. J. (2013). Towards dynamic trot gait locomotion: Design, control, and experiments with Cheetah-cub, a compliant quadruped robot. The International Journal of Robotics Research, 32(8), 932-950.

[5] Witte H, Hackert R, Ilg W, et al. (2003) Quadrupedal mammals as paragons for walking machines. In: Proc AMAM – Adaptive Motion in Animals and Machines, pp. TuA-II-2.1–TuA-II-2.4.

[6]Witte H, Hackert R, Lilje K, et al. (2001) Transfer of biological principles into the construction of quadruped walking machines. In: Proceedings of the Second International Workshop on Robot Motion and Control.

[7] Zhang ZG and Kimura H (2009) Rush: a simple and autonomous quadruped running robot. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 223(3): 323–336. DOI:10.1243/09596518JSCE668.

Appendix

Statement of contribution:

Student 1: (name, matric.) **Song Qingfeng A0119083H**

Student 2: (name, matric.) **Tu Fangwen A0109982U**

Student 3: (name, matric.) **Wang Kangli A0055862Y**

A. Joint Work:
   Joint research work about the modelling, control and simulation of quadruped dog. Joint work in project report write up.

B. Individual Work:
   Student 1: (chapter 2 and 3) Literature research on the quadruped dog leg mechanical design. Research on the modelling of a quadruped dog leg. Built up the kinetic and dynamic model of the quadruped dog leg.

   Student 2: (chapter 4) Research on the simulation and animation of the quadruped dog. Built up the ADAMS and Matlab simulation model for the quadruped dog based on the model. Gait planning simulation with PD control.

   Student 3: (chapter 5 and the rest) Control of quadruped dog robots. Designed and simulated computed torque method based control law and adaptive control based on the model, and help Qingfeng collection information for the literature research part.

We all agree that the statements above are truthful and accurate.

Signatures and Names:

SONG QINGFENG	TU FANGWEN	WANG KANGLI
		
(Student 1)	(Student 2)	(Student 3)