Proceedings of ASIMMOD 2009, January 22-23, Bangkok, Thailand

Hexagonal Structure Hexapod Robot: Developing a Method for Omni-directional Navigation

[1]Md. Masum Billah, [2]Mohiuddin Ahmed, [1]Md. Raisuddin Khan and [1]Soheli Farhana
[1]Faculty of Engineering, International Islamic University Malaysia (IIUM), Jalan Gombak,
53100 Kulala Lumpur, Malaysia.
[2]Faculty of ICT, International Islamic University Malaysia (IIUM), Jalan Gombak,
53100 Kulala Lumpur, Malaysia.

Abstract: This paper presents an optimal gait generation algorithm for a hexapod robot with hexagonal structure. Typical body structures of hexapod robots are analyzed. Various constraints like stability, foot force, leg workspace and leg collision that affect the walk, are taken into consideration to maximize the stroke of a leg during direction phase by making it pass through the center of workspace. The gait generated extracts maximum stroke length subject to the constraints, with the legs on ground during locomotion. Finally, an algorithm is developed for omni-directional navigation for a hexapod robot.

Keywords: Optimal gait generation; walking machine; Hexapod

INTRODUCTION

The better rough terrain mobility of omni-directional walking machines over wheeled vehicles has generated a significant research interest in the development of walking robot. Control duty for a walking robot involves leading execution of a command for walking, without losing stability and continuity of motion. Due to the complexity involved, the control duty has been split into levels such as higher-level controller and lower level controller. The higher level interprets the operator commands to actuator motions for the specified body motion. These actuator motions are then converted into actual movements of legs and body by the lower level controller. In order to make the machine omni-directionally navigated, it is required to make the higher-level controller more competent. This paper presents the development of higher-level controller algorithms for omni-directional walking of a hexapod robot.

Some problems of free gait generation were identified by researchers in various ways. The geometric approach[1] of varying gait parameters like stroke, duty factor relies on the modeling capabilities thus restricted to a particular direction. A constraint based approach can be useful for the robot and its environment taken into consideration while formulating the constraints. The behavior based approach[2] integrated with planning has its capability dependent upon the reaction time of the machine, which may be critical when a heavy vehicle is moving on a highly unstructured terrain where reaction time needs to be very small. Graph search and hierarchical approach[3]-[5] rely on a rule base to trim several less promising branches at a decision making stage.

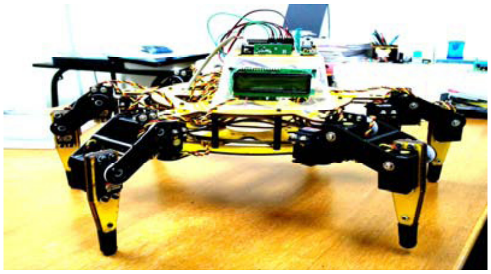

Fig. 1: The walking hexapod robot

DESCRIPTION OF THE WALKING ROBOT

The walking machine under consideration, hexapod robot, is shown in Fig. 1. The walking machine has six legs arranged axe-symmetrically

Corresponding Author: Md. Masum Billah, Department of Mechatronics Engineering, Faculty of Engineering, International Islamic University Malaysia (IIUM), Jalan Gombak, 53100 Kuala Lumpur, Malaysia, E-mail: liton_aub@yahoo.com.

around a vertical axis passing through the center of the body. It stands 6 feet tall and is having a mass of 1 kg. Each leg has three degrees of freedom vertical, horizontal, and twirl. The limits on leg movements in these directions define the three dimensional workspace in which direction the leg can be moved. Our task here is to design a higher level controller strategy for effective and efficient omni-directional walking of the hexapod robot.

The higher-level controller optimizes the way to move the feet (the servos) in order to achieve the motion of the body.

The higher level controller was developed[6] [7] which are capable of generating straight line motion along any direction motions about any point with a free gait. For maximizing stroke of the legs in each case, it determines optimal foot locations for straight line and circular motions of the body. The algorithm is also capable of generating total motions for any directional paths concatenated with hexagonal structure.

The above strategy is applicable only for two types of body orientations. One in: which the body orientation is kept fixed and the-other in which the body orientation is always in the direction of the tangent to the path traced by the body center. The more general motion of the body should allow the body orientation to change independent of the path of center of mass.

THE CENTRAL IDEA

When a tripod is to be placed on the floor to start a stance phase, we need to decide where exactly each foot of the legs has to be placed within its workspace. The importance of this decision is that some choices of the foot location lead to short strokes, forcing the tripod to be lifted very soon after it starts the stance and consequently forcing the tripod in the air to be brought down quickly. Foot placements, which maximize stroke, are useful, but difficult to determine exactly, especially when the gait is not regular. The main contribution of this work is an algorithm for determining foot placements, which lead to relatively large strokes. We do this by ensuring that each foot passes through middle of its workspace during the stance phase. Fig. 2 illustrates this idea.

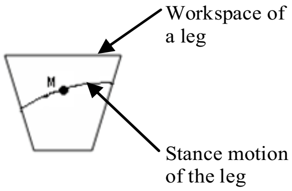

Fig. 2: Stance motion passing through mid position of workspace

The middle position of the workspace is shown as point M. We ensure that the path of the foot during stance passes through this point at all stance motions. The extent of stance motion on either side of pint of M is determined to maximize the stroke, subject to constraints like stability margin, foot force limit, workspace limits and avoiding collision amongst legs.

GAIT GENERATION

We assume that the x and y (horizontal) components of body center and the orientation, θ angle have been specified as functions of a dummy parameter. A useful dummy parameter is the distance along the curve representing body motion in the three dimensional space of x,y,θ. At each point of motion, we have to ensure that stability margin is sufficient, force on each foot is not beyond its limit, each foot is within its workspaces, and no two legs are colliding.

When starting, we assume that one tripod is in stance phase, with each foot at the mid point of its workspace. We first determine all the stance motions of the two tripods, and then join the adjacent stance motions with smooth transfer motions of the feet. The crucial algorithm is that of determining the touch down foot locations for a tripod for starting a stance. The basis of this algorithm is to estimate the instance at which the feet of the tripod arrive at their respective mid points, so that the touch down points can be obtained by working backwards in time. The estimate mentioned above is modified to ensure that there is sufficient overlap between two tripods. This uses the information about the instance at which the previous tripod is lifted. The algorithms for touch down and lift off are described in detail below. The dummy parameter used for specifying body motion is L. We consider discrete values of L to make the algorithm faster.

A. Algorithm for determining touchdown position

The problem of determining the touchdown position can be solved iteratively as follows.

1. Let the value of L at which the current tripod (assume 2-4-6**)** is getting lifted be Le. Let the specified overlap between the stance motions of two tripods be (in terms of number of intervals n_o of L) n_o ▲L. Then the value of L at which the next tripod has to be placed is L1 =Le – n_o ▲L.
2. Choose an estimate Lm, the instant at which the tripod to be put down (1-3-5) will reach its mid stance position.
3. Determine the motion of feet of the tripod 1-3-5 from the respective workspace midpoints at Lm, backwards in time, till the first occurrence of violation of any of the constraints (Fig. 3).

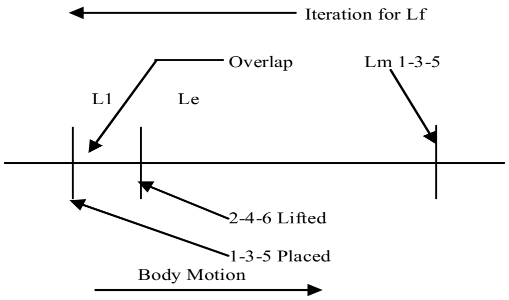

Fig. 3: Representation of hack iteration to obtain the touchdown position L1.

4. Let the earliest violation of constraints be at Lf. We calculate the correction required on the initially selected Lm as DL = Lf-L1. This is the amount by which the calculated touchdown position is different from the required value.
    - If DL is negative and less than DLT in magnitude (a tolerance), no adjustment is required on Lm and proceed to step 5
    - If DL is negative and greater than DLT in magnitude, more stride is possible than our initial guess of Lm, and we can increase Lm by DL and repeat step 4.
    - If DL is positive, new smaller Lm is calculated as Lm = Lm-DL and step 4 is repeated.

We now have the correct Lm such that when the tripod touches down at L1, no constraint is violated till Lm. The sequence of foot positions, generated from mid stance to touchdown, is stored. Note that for forward body motion, the legs move backwards in body coordinate frame, so while back iterating for Lf, the legs move forward for checking the limits.

B. Algorithm for determining the liftoff position

Here we determine how far a particular support tripod can move the body forward. Starting from mid stance position Lm, we find an instance where first violation of any of stability constraint, workspace constraint, foot force constraint or collision constraint occurs.

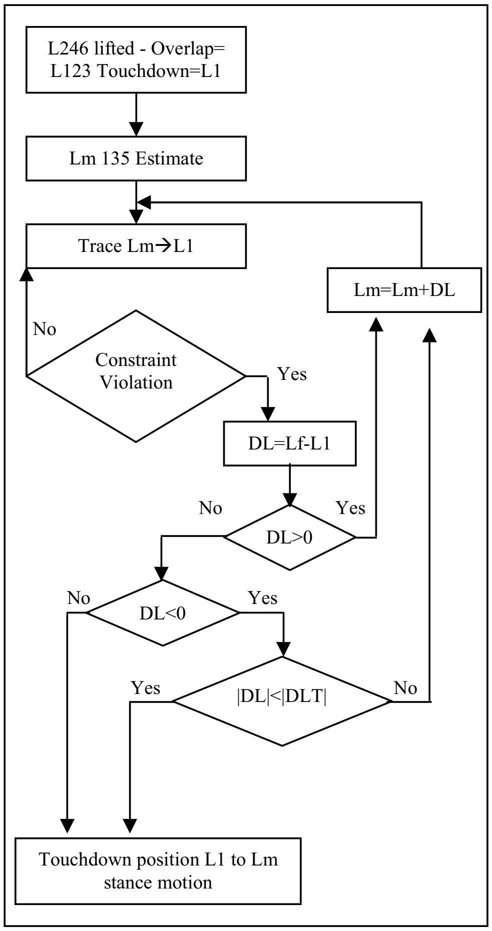

Fig. 4: Flow chart for algorithm of determining touchdown position

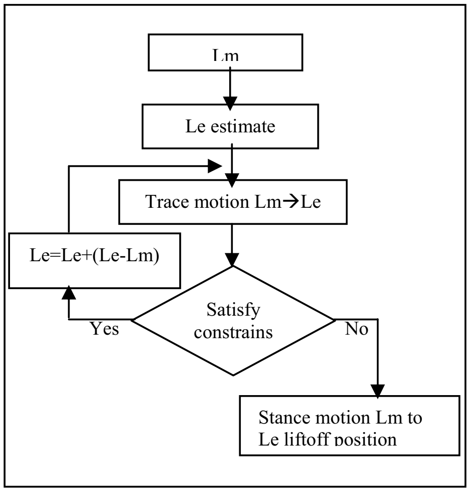

Fig. 5: Flow chart for algorithm for determining Liftoff position

This instance would be marked as a liftoff instance Le, for that support pattern. Thus this algorithm determines the value of the parameter L at which the tripod has to be lifted and the stance motion from mid stance to liftoff.

The above two algorithms provide the stance motion for the body, where each leg passes through its middle position of its workspace during the stance phase. Determination of positions of the feet of a tripod during stance is based on the known body motion and the position of the three feet in world coordinate frame at touch down. When the body undergoes a known displacement from its touch down position, the stance feet remain fixed at known positions in the global frame. Coordinate transformations are then used to obtain the positions of the stance feet in the robot's body fixed coordinate system. We now describe how the constraints are calculated.

C. Determination of stability margin

Stability margin is defined as the least distance from the vertical projection of center of mass on ground to the convex hull formed by the feet on ground. We compute the distance of center of gravity from all sides of support pattern. The minimum of the three distances is the stability margin. This should be more than a specified minimum stability margin.

D. Determination of reaction forces of stance feet

The design of the walking robot and its feet is based on the condition that a leg would be able to bear a weigh equivalent to 1kg anywhere within its workspace. Given the set of tripod positions with respect to body frame during stance motion and the mass and payload of the machine, we, calculate the reaction forces coming on the feet. A support motion remains feasible if the reaction forces of the feet are within the specified limits.

E. Checking foot workspace limits

The legs of the machine need to be in their respective workspaces during stance as well as transfer phase. We define certain workspace for each leg depending on their kinematics constraint and examine whether the leg remains within that by a margin called workspace margin. Limits are put on all three direction movements of the leg, radial, swivel and vertical. The foot position is transformed into its leg coordinate frame in radial coordinates and is examined to satisfy the limits.

F. Algorithm for determining collision among legs

This algorithm detects whether there is any collision between two neighboring legs while following a specific path with certain sequence of foot positions. For checking collision between legs, we examine some critical points of a leg. We identify the outermost points of a leg depending upon whether the leg is stretched or folded. The potential collision situation can occur when a leg is touching down and its neighbor is ready to be lifted, i.e., in the overlap portions of the motion. We follow following steps

- Find out whether a leg is folded or stretched i.e. examining the radial distance of thigh and the foot tip.
- Express the critical points of a leg in its neighbor's leg reference frame as shown in Fig. 7, for the global scene as Fig. 6.
- Check whether the critical points are outside the collision margin polygon PlP2P3P4 as shown in Fig. 7.

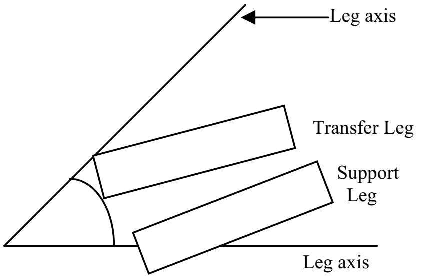

Fig. 6**:** Global scene of legs at the critical moment

Thus a touchdown position is collision free if there is no collision for all three support legs and corresponding brackets.

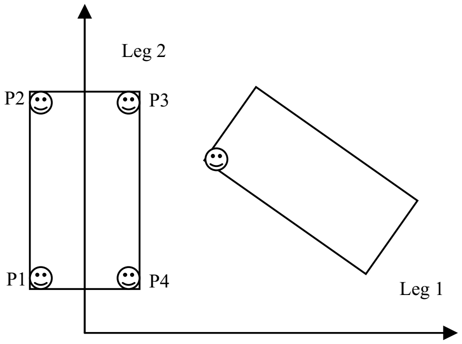

Fig. 7: Points on a leg expressed in its neighbor's leg reference frame

G. Determination of transfer motion

After determining all stance motions, transfer motions of feet are determined as smooth curves joining a liftoff to next touch down. The transfer paths of feet have to be within their respective workspaces. We have the Limits on radial, swing and vertical movements of the legs due to workspace and kinematics' constraints. We know the xyz coordinates of a leg at the time of liftoff and touchdown; we convert this position in respective radial coordinates, and fit a curve between these two radial positions, which will represent the radial leg motion in transfer phase. Similarly we fit a curve for the swing motion, keeping position and velocity continuity at liftoff and touchdown. While generating the transfer motion in vertical direction we keep an extra constraint that the leg should get lifted by a specified amount. This transfer motion generated in radial coordinates is convened to xyz coordinates and is checked motion thus we have a transfer motion with minimum of swing, radial and vertical movements. Combined motion of all the transfer legs constitutes the transfer motion for the tripod.

At the beginning and at the end of motion, we assume that the tripod in transfer phase has feet at their mid stance raised position.

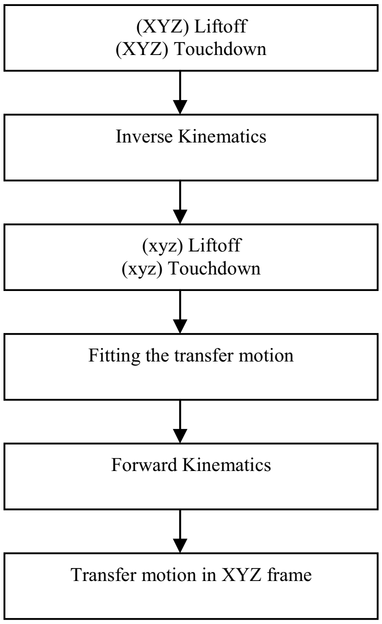

Fig. 8: Flow chart for transfer motion generation

The above sections described the planning of leg trajectories in terms of position and distance traveled. This motion is in terms of the parameter L of the walk. Now we need to convert this motion with respect to time, in order to determine terms like speed and acceleration of travel. An earlier 'developed algorithm[7] for controlling speed of motion of the machine, determines the exact variation of parameter L with respect to time so that the potentiometer limits are satisfied. Fig. 9 gives the flow chart of gait generation algorithm.

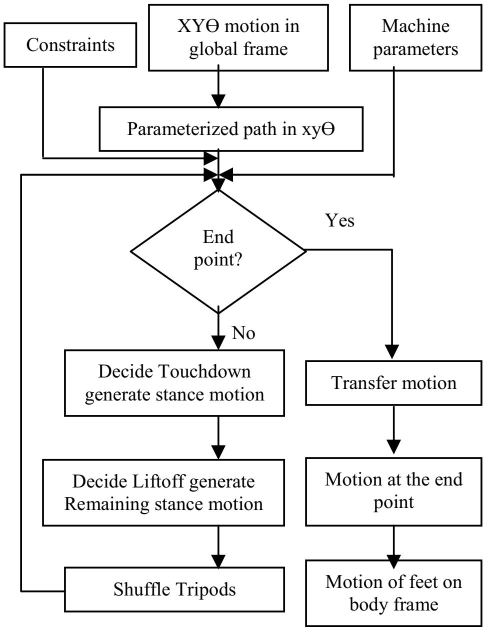

Fig. 9: Flow chart of gait generation algorithm

CONCLUSION

The problem of optimal gait generation for a six legged walking machine, hexapod, is addressed here. An algorithm, which generates near maximal stroke tripod gait, has been developed for walking on regular terrain. Limits on minimum stability margin, maximum foot force, foot motion and collision between adjacent legs are considered for generating the gait. The algorithm is capable of generating leg trajectories for complex paths with body allowed to turn while walking. The algorithm can be used with minor modifications, for generating regular gaits like wave gait and for free gaits, and also for walking on inclined planes and steps. Walking on irregular terrain would require some substantial extensions.

REFERENCES

1. Kumar V. R. and Waldron K.J., Adaptive Gait Control for a Walking Robot, Journal of Robotic Systems, pp 49-76.
2. Wettergreen D., Thorpe C., Gail generation for Legged Robots, Proceedings of the IEEE International Conference on intelligent Robots and Systems, July 1992.
3. Pal P.K. and Jayarajan K. Generation of Free Gaits- A Graph Search Approach, IEEE Transactions on Robotics and Automation, Vo17, No. 3, June 1991 pp 299-305.
4. Ching-Long Shih and Klein C.A., An Adaptive Gait for Legged Walking Machines over Rough Terrain, IEEE Transactions on Systems, man and cybernetics. Vol23, No. 4, August 1993, pp 1150-1155.
5. Mcghee R.B., lswandhi G. I. Adoptive Locomotion of Multi-legged Robot over Rough Terrain, IEEE Transactions on Systems, man and cybernetics, Vol SMC 9, No. 4, April 1979, pp 176-182.
6. Nilesh S. Brahme and K. Kurien Issac, Gait generation for LI Six legged Walking Robot, Proc. of the 10th National Conference on Machine and Mechanics, 209-216, 2001.
7. Amit P. Gabale. K. Kurien Issac, Consideration of Gait Transitions and Extended Feet in the Determination of Foot trajectories for a Six Legged Walking Robot, proceedings of the 11th World Congress in Mechanism and Machine Science, Tianjin, China, August 2003.