

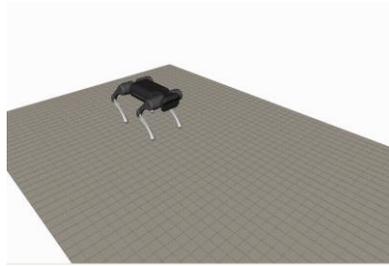
Gait Planning

Objective

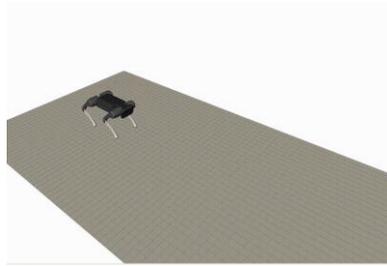
This is a gait planning guide that provides a *very brief* overview of what gaits are, how they are designed, what purpose do they serve, what are the essential aspects to them, and how they can benefit us. This guide will discuss gaits in general but emphasis will be given on quadruped robots, hence, the reader is advised to keep that in mind. Moreover, this guide is intended to be available for the general public, hence, some basic concepts are discussed, advanced concepts are briefly touched upon but may provide an interesting read for advanced students and researchers who are unfamiliar with gait planning.

Introduction

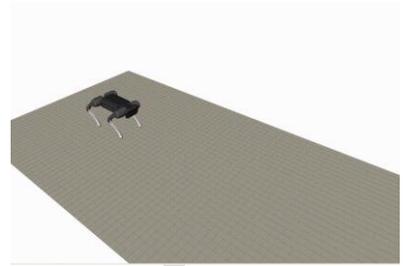
Quadruped robots are one of the most advanced and sophisticated robots available today. Its name *quadruped* stems from four-legged animals and as these robots have four legs, hence, *quadruped robots*. Different legged robots are attributed according to the number of legs they have, so a humanoid type robot with two legs would be classified as a bi-pedal robot and a robot with eight legs would be classified as an octo-bot. The motion patterns by which these robots move are known as *Gaits*. A legged entity has motion patterns, so for us, humans *walking* would be a gait, *running* and *hopping* would be another, and so on. This is one of the core aspects of *legged robots*. Legged robots have many attractive applications such as the potential ability to substitute the labor workforce, exploring inhospitable zones, search and rescue operations, or simply as companions. However, there is a catch, which is that any type of legged robot presents a significant challenge over its traditional wheeled and tracked counterparts. One of the major reasons being that for any type of simple movement, such as *walking* the robot requires intensive physics calculations taking into account moment of inertia, foot contacts, applied load, path trajectory, obstacle avoidance, environment, terrain, friction, inverse kinematics, just to mention a few. Once these calculations are performed, they are then required to be transmitted in real-time and computed into low-level commands for the actuators to move the robot.



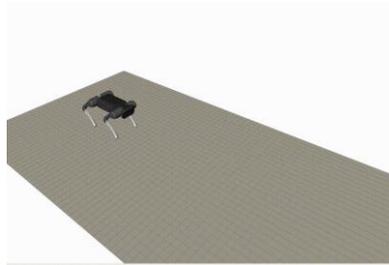
Quadruped robot turning with pace gait



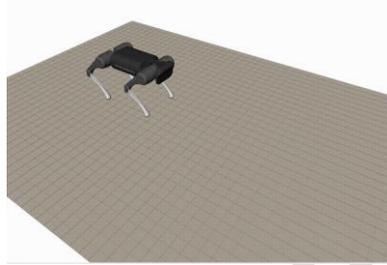
Quadruped robot with hop gait



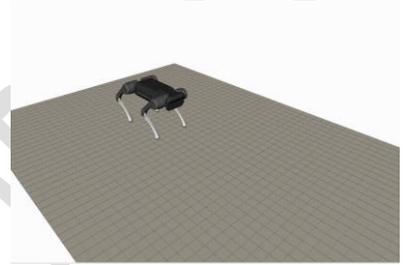
Quadruped robot with walk gait



Quadruped robot with trot gait walk gait



Quadruped robot reversing with gait



Quadruped robot turning with canter

Motivation

A question arises that if such legged robots are so complex and difficult to use then why to bother with it when existing and proven technologies like autonomous wheeled vehicles and *Unmanned Aerial Vehicles* (UAVs) are available. The answer is simple, legged robots, provide versatility, agility, and strength. Its versatility and agility enable it to pass obstacles that are otherwise un-passable for wheeled vehicles, e.g. traversing over a fence while maintaining the ability to drag and carry large amounts of heavy loads which is not possible for UAVs.

Obstacles

Legged robots are able to move over maximum ground clearance terrain by stepping over them, moreover, it is able to clear its own height and more by jumping, which is more than a tracked vehicle or a wheeled robot.

Slippage

Wheeled robots are susceptible to slippage in soft terrain such as sand or mud, whereas due to the independent actuation of each leg of a legged robot, it is able to bypass this difficulty.

Terrain

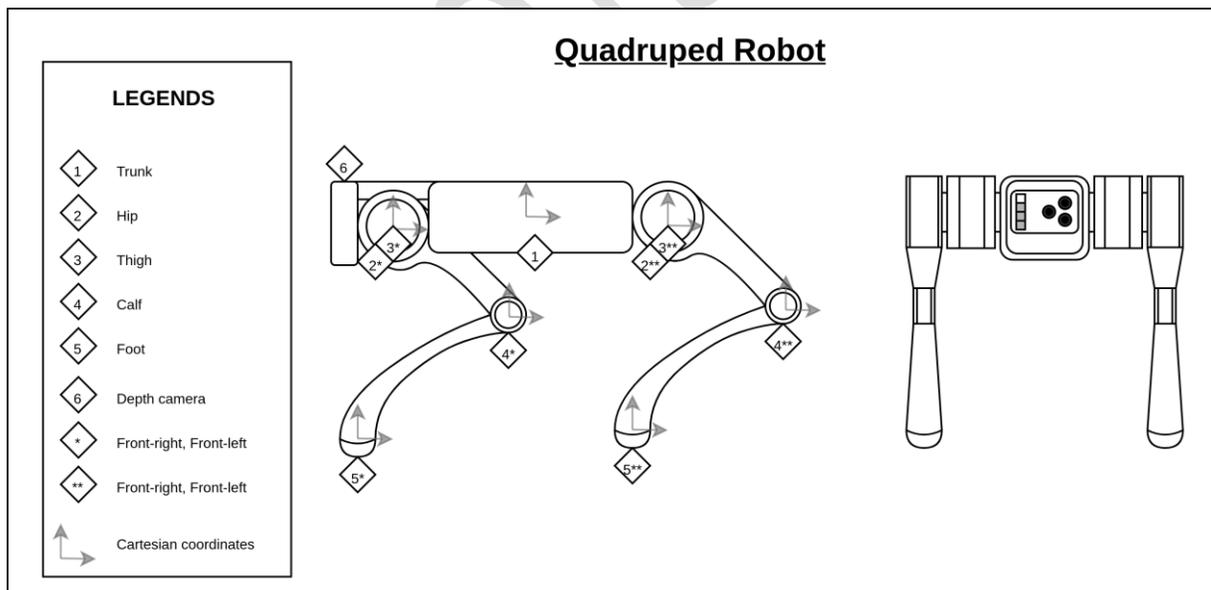
The natural terrain is a difficult domain for robots in general; for wheeled robots, this is a nightmare. Moving in a sandy area, or through a forest require augmentation for wheeled robots. They always require continuous terrain to move, however, for a legged robot, they are able to traverse most types of terrains without any augmentation and are able to move in discontinuous terrains like hopping between the rocks in a mountain.

Speed

In flat and well-constructed terrain, a wheeled robot is easily able to surpass a legged robot, on the flip side, in any terrain other than a well-constructed terrain, legged robots move significantly faster, with less energy consumption, and less damage to the environment. A study conducted on the speed of tracked vehicles wheeled vehicles, and four legged animals show a surplus speed of 312 for animals, which means that with a well-developed legged robot, such speeds can be acquired and surpassed.

Components for a Quadruped Robot

The following diagram illustrates some of the common and essential components that quadruped robots have:



Content

With a brief overview of quadruped robots, we get back to the main topic `gait planning`. The content that this guide will briefly cover is:

- Gait design
 - Speed
 - Gait diagram
- Stability
 - Static stability
 - Dynamic stability
- Periodic gait generation
 - Continuous gaits
 - Discontinuous gaits
 - Turning gaits
 - Path tracking gaits
 - Comparison of continuous and discontinuous gaits
- Non-periodic gait generation
 - Terrain model and adaptation
 - Leg sequence planner
 - Foothold planner
 - Body motion planner
- Implementation of gaits
- Additional Material References
-

The reference material used for this guide has been adapted from Santos [2].

Gait Designs

The first documented walking mechanism was built in 1870 by Chebyshev which had four bars that rotated around a designated fixed axis. This mechanism worked by alternating the bars, moving each leg between a stance and a swing phase. However, this mechanism was only able to walk on flat surfaces, and due to its fixed duration of alternating between a stance and a swing, it only achieves static stability (Static stability means that an entity remains stable without falling or tipping over without the need of external intervention). The disadvantage of static stability is that a legged robot is unable to operate and move at high speeds, severely limiting its usability. Reiterating on the definition of gait in a more formal way, gaits are the sequence of motion required by a mechanism to move both the legs and body of a robot. The

mechanism may be of any kind such as a simple actuator or hydraulic actuator. Actuators that are not revolutes may require additional components to convert their motion into a rotary motion, as those are the most commonly used form of actuation for legged robots. Gait designing and gait generation are based on mathematical formulas which depend on the length of the leg, leg phase, duty cycle, etc. For adapting to irregular terrains, the parameters are to be updated on the fly and require sensors that capture the terrain ahead so that footholds could be selected as well as an appropriate gait.

Speed

An important factor to keep in mind when thinking about gaits is the speed that a legged robot is capable of moving at. This can be represented in a general form:

$$V = \frac{R}{\beta} (1 - \beta) \tau$$

$$V = \text{Speed} \tag{1}$$

$$R = \text{Leg stroke} \quad \tau = \text{Leg} \tag{2}$$

$$\text{return time} \tag{3}$$

$$\beta = \text{Duty factor} \tag{4}$$

$$\beta = \beta_n = \frac{3}{n} \tag{5}$$

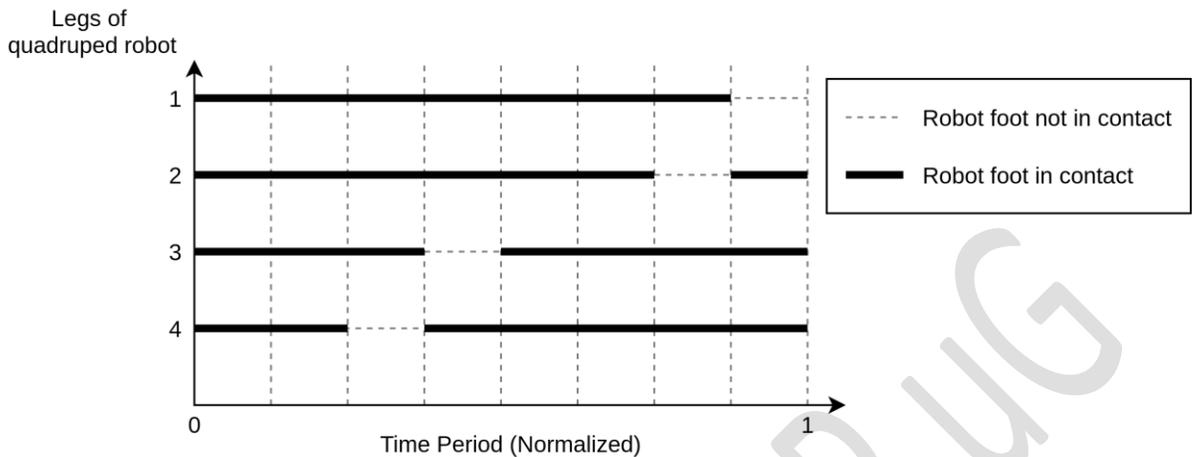
$$\tag{6}$$

$$n = \text{Number of robot legs} \tag{7}$$

Note: Specialized gaits have more intricate velocity formulas taking into account the duty factor as well as locomotion phase factor.

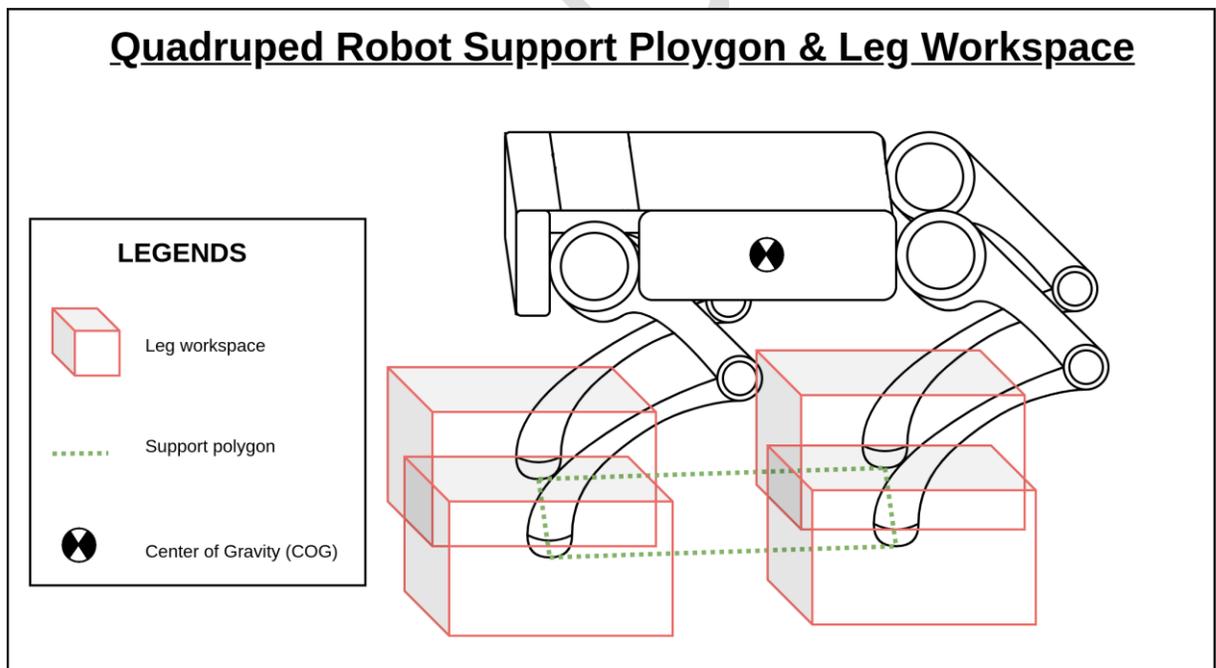
Gait diagram

Gait diagrams illustrate the movement patterns of legged robots. In it, the *x-axis* depicts the time for a gait sequence, and the *y-axis* depicts the number of legs in the given time sequence. The continuous lines represent the fact that the robot's foot is in contact with a surface whereas the dashed line represents the robot's foot not being in contact with the surface.



Stability

Stability is one of the core challenges in legged robotics. They introduce a new dimension of problems that need to be solved. Fortunately, researchers have come up with ingenious ways of ensuring the stability of these robots. We will be looking into the intuition of static and dynamic stability which are critical to take into consideration when working with a legged robot.



Static Stability

Starting off with static stability, static stability is the ability of a robot to remain stable when not in motion. This stability is judged by a stability measurement criterion. There have been a number of stability criteria proposed over the last century for legged robots such as *Static Stability Margin* (S_{SM}), *Longitudinal Stability Margin* (S_{LSM}), *Crab Longitudinal Stability Margin* (S_{CLSM}), and *Energy Stability Margin* (S_{ESM}). A more relevant and advanced criterion is the *Normalized Energy Stability Margin* (S_{NESM}) static stability criterion which takes into account the dynamic effect, dynamic model, kinematic model, and the *Center Of Gravity* (COG) of the robot. It can be defined as the minimum potential energy required for a robot to stumble given a support polygon where a support polygon is a region in which the COG of a robot must stay in for a robot to be statically stable. More in-depth information is available by Messuri [3]. The formula for S_{NESM} is:

$$S_{ESM} = \min_i (mgh_i) \quad n_s = \text{Number of} \quad (8)$$

$$\text{supporting legs } i = \text{Number of} \quad (9)$$

$$\text{supporting legs} \quad (10)$$

$$m = \text{Mass of robot} \quad (11)$$

$$g = \text{Gravity constant} \quad (12)$$

$$h_i = |R_i|(1 - \cos\theta)\cos\psi \quad \text{Height of tumble} \quad (13)$$

$$S_{NESM} = S_{ESM}/mg$$

Dynamic Stability

Similar to static stability, dynamic stability is also required for legged robots. It as well has several types of stability criteria. One example that extends on the S_{NESM} is the *Normalized Dynamic Energy Stability Margin* (S_{NDESM}). In it, there is the addition of kinematic forces on the robot, given by:

$$S_{NDESM} = \min(E_i) \quad (15)$$

$$E_i = mg|R_i|(\cos\theta - \cos\varphi)\cos\psi + (F_{Ri} \cdot t)|R_i|\theta + (M_{R_i} \cdot e_i)\theta - \frac{1}{2}I_i\omega_i^2 \quad (16)$$

$$= \text{Orthogonal vector to the support polygon} \quad (17)$$

$$F_{Ri} = \text{Ground interaction without gravity} \quad (18)$$

$$I_i = \text{Moment of inertia } \omega_i = \text{Angular} \quad (19)$$

$$\text{velocity around COG} \quad (20)$$

$$\theta, \varphi, \psi = \text{Rotation around the } i \text{ axis} \quad (21)$$

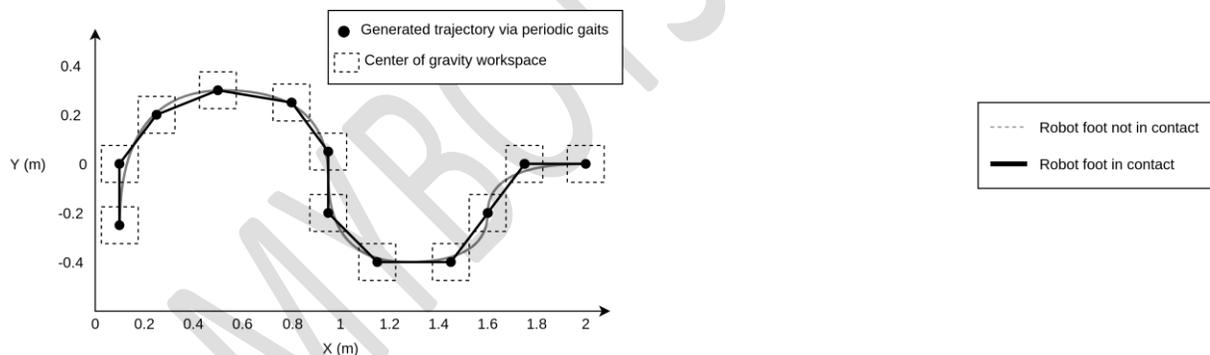
It is important to keep in mind that the stability criterion for both static and dynamic stability is not fixed and mainly depends on the application and requirement of the robot. New criteria are regularly proposed and continuously change, hence it is vital to know the factors that influence these criteria. Several of the latest criteria take into account the torque of the motor as well as a dynamic payload which has not been mentioned here.

Periodic Gait Generation

In periodic gait generation, events are sequenced out for the time and placement of each foot of a legged robot required to move its body. Each *event* corresponds to either a foot being placed or a foot being lifted. If two *events* occur at the same time, they are called **singular gaits**. All others are **non-singular gaits**. Singular gaits are important for the static stability of a robot and are used for locomotion.

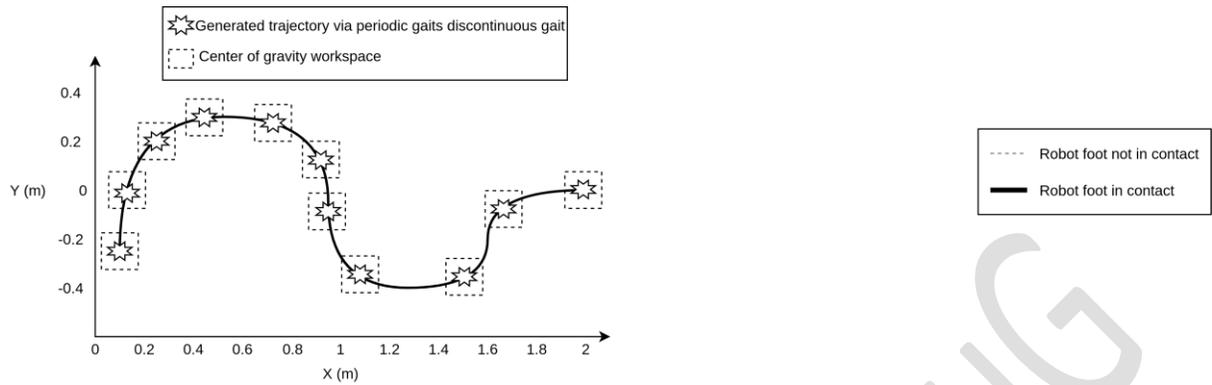
Continuous Gaits

An important singular gait to know about is the *creeping gait* or the *crawl gait*. This gait ensures static stability by restricting the number of foot contacts to $n - 1$ that a legged can have at any given moment due to which it is a staple gait and used in many of the modern legged robots today. Another type of periodic gait is the *wave gait*. It works on the principle that any pair of legs should move half a cycle out of phase.



Discontinuous Gaits

In these types of gaits, a sequential motion is followed in which the robot is propelled forward by legs that are in a support position. What this means is that at any given moment, only one leg of the robot would be in movement whereas all other legs would be in contact with the terrain. The support legs that are in contact with the surface move the robot forward and the leg that is reaching its kinematic limit is lifted and shifted forward.



Turning Gaits

The gaits that have been discussed are gaits that move the legged robot in a straight trajectory and to perform a turn they combine gaits in a way that the stability margin is not violated. For turning a legged robot about its own vertical axis, two primitive gaits are typically used are the *circle gait* and the *spinning gait*. Similar to the gaits that generate motion for straight paths, turning gaits generate foot contacts that do not violate the stability margin. This itself requires more computation compared to straight paths and is dependent on the required turn angle.

Path Tracking Gaits

For periodic gaits, path tracking is achieved by switching with pre-determined gaits over a trajectory. The trajectory is ideally generated in a way that the stability margin is adhered to and the COG of the robot does not leave the region in which it could become unstable. The same is the case for the discontinuous gait.

Comparison of Continuous and Discontinuous Gaits

Comparison table for continuous and discontinuous gaits

Continuous Gaits	Discontinuous Gaits
Smooth motion	Improved stability
Simple Implementation	Simple Implementation Higher maximum speed
	Jerky motion

Non-periodic Gait Generation

Non-periodic gait generation is as the name suggests, a method of generating non-sequence gaits for moving a legged robot. The necessity of non-periodic gaits arose due to the inability of periodic gaits to traverse dynamic and complex terrains. As an example when we humans traverse a rocky area, we do not follow a fixed gait, instead, commonly we scan the area for appropriate footholds and move each of our limbs in non-periodic gaits to reach our desired destination. In short nonperiodic gaits allow *true* adaptability of legged robots to move in rough terrain and are typically known as *free gaits*. Free gaits primarily depend on 3 aspects: The first is the trajectory, the second is the terrain, and the third is the robot's state. There are two main methods for generating free gaits, *ruled-based* and *search-based*.

Ruled-based gaits can be adapted from biological organisms or can be manually designed whereas in search-based methods random footholds are computed to generate a trajectory. For ruled-based gaits, excessive simplification is required for models due to complex formulations and as for searchbased gaits, several restrictions have to be placed so that the searched footholds are selected in a way to ensure stability. This brings us to the main challenge in non-periodic motion which is to balance planned actions with reactive behavior for the robots. Important aspects to keep in mind when designing a rule-based gait or a search-based gait are as follows:

Terrain Model and Adaptation

Terrain realization is very important especially for legged robots due to the kinematic constraints on them. When computing trajectories over terrain it is important to keep in mind the workspace for each foot of the legged robot, as too deep ditches may be un-adaptable, hence, these regions should be considered special regions and the same goes for too high platforms. These special regions can then be designated as forbidden regions or may be subjected to special trajectory planning for traversal. In all cases, if a robot is required to move autonomously on these terrains, then a sensory system that is able to detect and differentiate the height of the terrain is a must.

Leg Sequence Planner

The leg sequence planner is responsible for determining the position of the foothold for the current time instant and the next time instant while maintaining the stability threshold. A methodology of applying this is by first selecting a gait such as the wave-gait and then adapting new footholds within this gait while not violating the stability, motion sequence, and kinematic constraints.

Foothold Planner

The foothold planner is responsible for determining the footholds for the legged robot. It checks several criteria, such as whether the foot of the robot is within the legs workspace, whether the foot is in a

forbidden region or a special region, collision amongst the legs, and whether the foot is above the maximum kinematic constraint.

Body Motion Planner

The body motion planner takes into account the path of the legged robots, and checks to see based on the legged sequence planner and the foothold planner if the COG moves out of its designated stable workspace.

Conclusion

In this article, we briefly went over the core aspects of legged robots' locomotion. We covered some of the standard periodic and non-periodic gaits, as well as looked into some of the free gaits. Prominent stability margins were discussed as well as the advantage and disadvantages of using a legged robot.

Commonly Gaits

Some common and useful gaits that are usually integrated into legged robots are:

1. Creeping\Crawl gait
2. Stride gait
3. Walk gait
4. Trot gait
5. Hop gait
6. Gallop gait
7. Pace gait
8. Bound gait
9. Amble gait
10. Canter gait

Implementation of Gaits

A few open-source packages are available in which you can try to integrate either periodic/nonperiodic gaits or free-gaits with your legged robot are:

1. [Towr](#)
2. [Champ](#)
3. [Free Gait](#)

Additional Material

Most of the material here has been adapted from Santos [2]. Some things that are vital but have not been covered are the *crab-gaits* and the *crab-angle* which is also required for the stability of a legged robot. As this cannot be discussed without going into heavy details, it has been skipped.

Material on this is available in [2].

References

- [1] Yin, C., Zhu, J., & Xu, H. (2009). Walking Gait Planning And Stability Control. Humanoid Robots. <https://doi.org/10.5772/6735>
- [2] Gonzalez de Santos, P., García, E., & Estremera, J. (2011). Quadrupedal locomotion. Springer.
- [3] Messuri, D. (1985). Optimization of the locomotion of a legged vehicle with respect to maneuverability. Ph.D. thesis, The Ohio State University.

Note:

Please email at salman.sohail@mybotshop.de in case you find any incorrect statements, facts, typos, or if you think some additional information should be added.