Legged Robotics

FIELD AND SERVICE ROBOTICS

DIETI. UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II
DIPARTIMENTO DI INGEGNERIA ELETTRICA E TECNOLOGIE DELL'INFORMAZIONE

www.prisma.unina.it

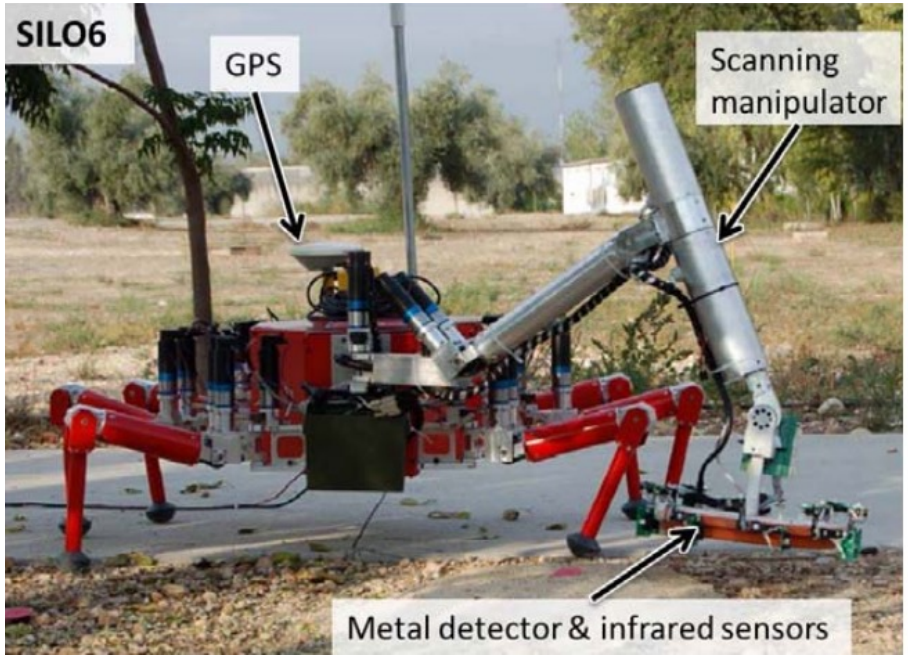
Motivation

- Approximately half of the Earth's land surface is inaccessible to either wheeled or tracked vehicles, legged robots can adapt themselves to the terrain irregularities varying their leg configurations
- Wheeled vehicles require paved surface for locomotion damaging the natural terrain
  - Legs of the walking machines create discrete footprints during locomotion causing smaller damage
- Inherent failure tolerance during statically stable locomotion
  - If one or more joints are damaged, the legged robots may be able to continue to move depending upon the number of healthy joints available.

- Exploration

- Forestry

- Humanitarian demining

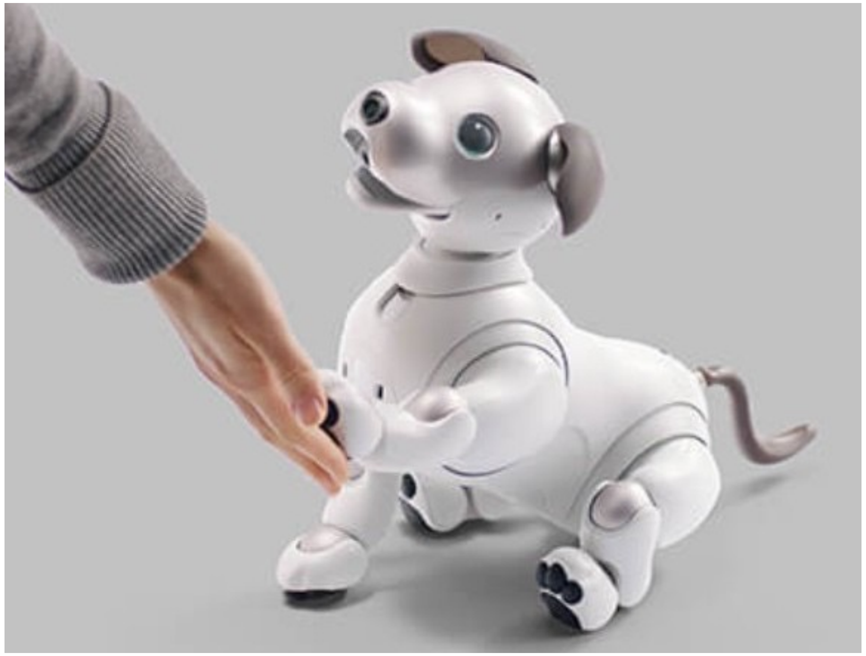


https://www.youtube.com/watch?v=IfULp2hqY8M

- Delivery

https://www.youtube.com/watch?v=U8Iey_wfo0I

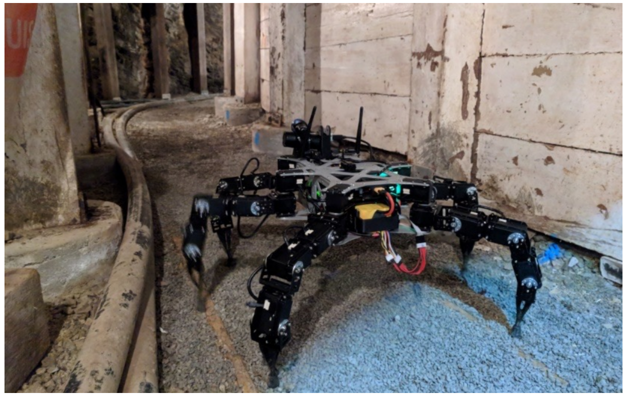
- Entertainment



- Other Applications:
  - Patrolling
  - Inspection in nuclear plant
  - Elderly care
  - Walking assistance
  - Pipe inspection
  - Rescue in disaster area
  - Constructing
  - …

- Oil and gas facilities
  - What is possible:
    - Visual inspection
    - Thermographic inspection
    - 3D mapping of existing plants
    - First intervention and monitoring
  - What is needed:
    - Power maintenance
    - Localization Capability

- Legged robots for inspection



- Inspection and maintenance in offshore sites, using visual and thermal cameras, microphones and gas detection sensors

https://www.youtube.com/watch?v=DzTvIPrt0DY

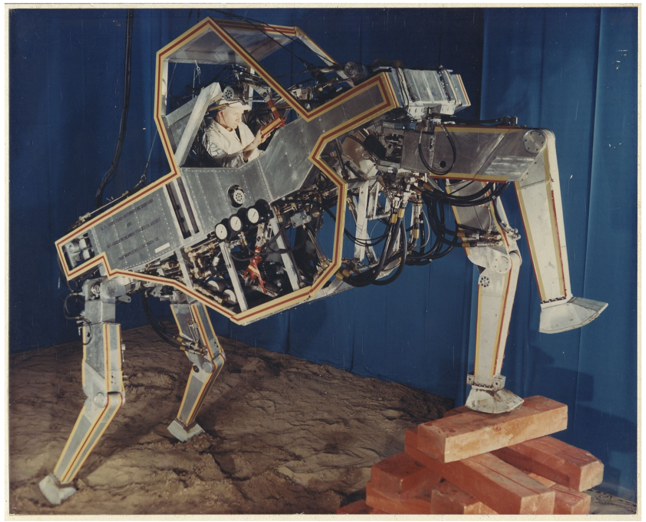
Cool (!)

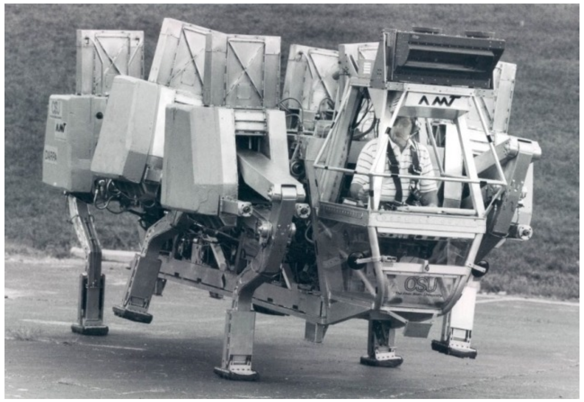
- … they are cool (!)

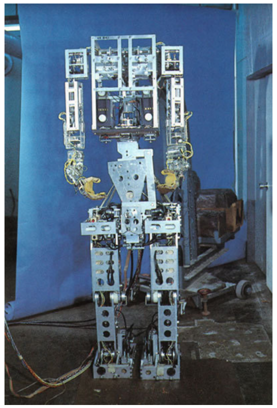
https://www.youtube.com/watch?feature=oembed&v=_sBBaNYex3E

- History:
  - Efforts to mobilize the artifacts by leg mechanisms have been mentioned in the mythology and ancient scripts from the ancient Greek, Indian, Egyptian, and Chinese civilization
  - Leonardo Da Vinci designed the first articulated anthropomorphic robot (Automa Cavaliere) in the history of western civilization
  - Before the advent of digital computers, legged machines could be approached only by electromechanical means, without the advantage of a feedback control

- Walking robotics research gathered a new momentum after the Second World War due to the new inventions in mechanisms, material science, electronics, control system, and computers
  - In 1968 the General Electric developed the first quadruped able to adapt its gait to the irregular terrain
  - In 1973, in Japan, the Waseda University built the first full-scale anthropomorphic robot in the world
  - In the mid 1980s, RobertMcGhee culminated his series of hexapod with the Adaptive Suspension Vehicle







- Transition to dynamic legged locomotion (hoping and running robots developed at the MIT by Marc Raibert) in 1980s

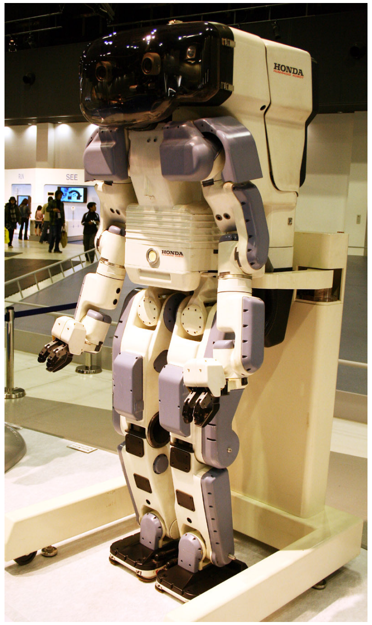
https://www.youtube.com/watch?v=XFXj81mvInc

- In late 1980s, Tad McGeer demonstrate that there is no need to have complete (or any) control to be able to walk dynamically and efficiently (passive dynamic walking)

https://www.youtube.com/watch?v=WOPED7I5Lac

Historical background

- The Honda Humanoid Robot project started in 1986, with the objective to create a robot coexisting and cooperating with humans. The latest version, Asimo, in 2000, it is a completely autonomous robot both in terms of processing capability and power. The previous version was the P2 humanoid in 1996



Historical background

- Sony began selling more than 150000 of its Aibo home companion robot dogs
- Marc Raibert founded Boston Dynamics after leaving MIT and unveiled Big-Dog in 2005

https://www.youtube.com/watch?v=5ifwGc-0mAY

https://www.youtube.com/watch?v=cNZPRsrwumQ

Historical background

- The progress over the last decades has been remarkable
- Profound questions have finally been answered, but other profound questions still have to be answered
  - How best to make them walk and run *efficiently*
  - Performance of legged robots needs to be improved in many ways
    - Energy
    - Speed
    - Reactivity
    - Versatility
    - Robustness
    - …

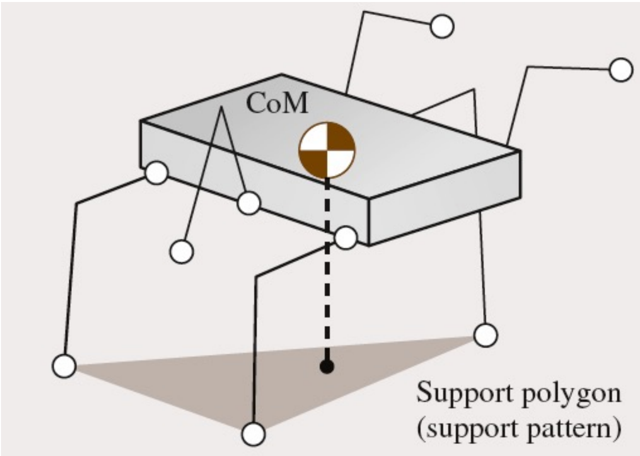
Research projects @ PRISMA Lab

www.weldon.unina.it

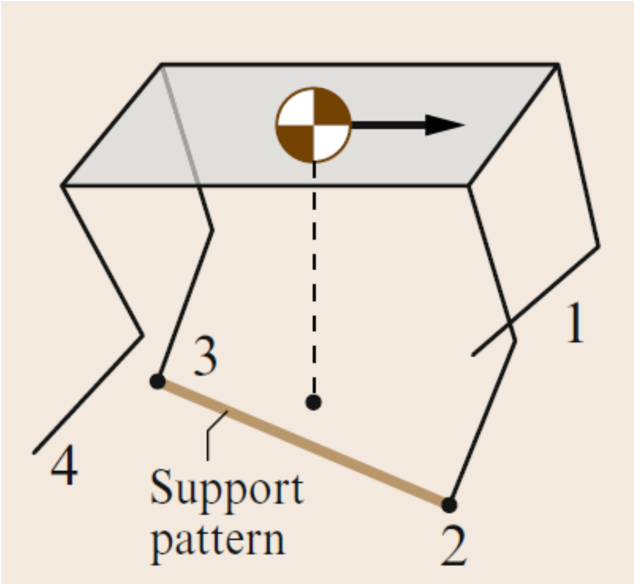
https://www.cowbot.unimore.it

Piano Nazionale
di Ripresa e Resilienza

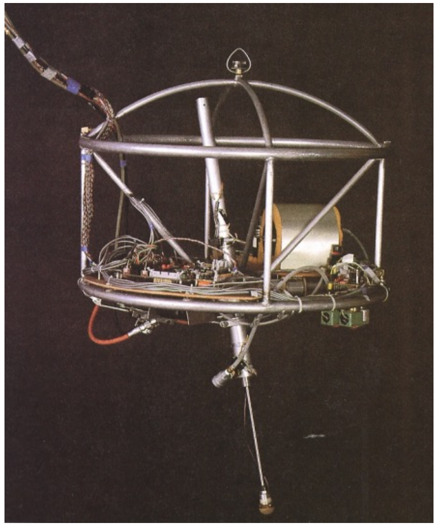
- Classification based on the stability of the gait
  - Statically stable gait : the robot control its locomotion such that the vertical projection of the center of gravity is always contained within the support polygon
    - The locomotion speeds of statically stable robots are quite slow
    - Six-legged and eight-legged robots are always statically stable
    - Quadruped robots are statically stable if they are in tripod gait: one leg in the air and three legs for support

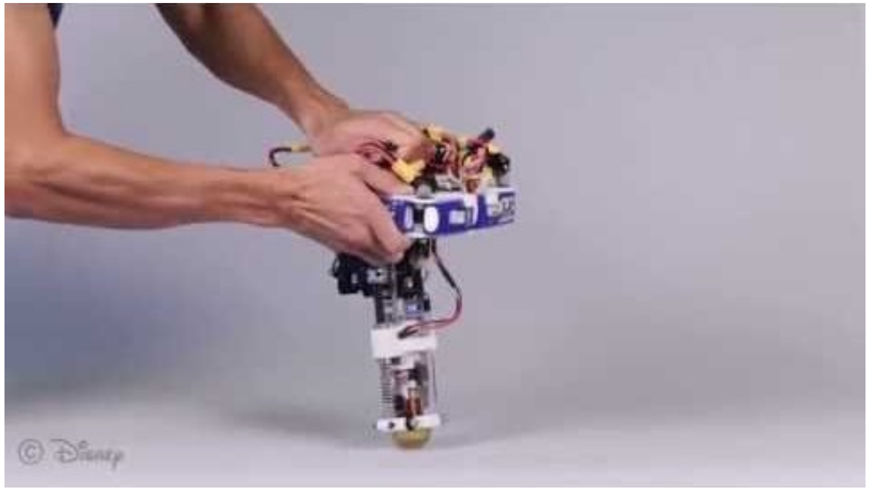


- Dinamically stable gait : the robot is not statically balanced during the locomotion
  - The support polygon has degenerated to a point or a line
  - The robots utilize dynamic forces and feedback to maintain control
  - Monopod hopping robot, biped or humanoid robot, and quadruped robots at high speed (e.g. trot gait) employ dynamic stability



- Classification based on the number of legs
  - Monopod robots: the locomotion is performed through hops , getting a dynamic stability
    - Potential application is the exploration of small celestial bodies, where legged and wheeled robots are not able to move successfully, due to the reduced local gravity



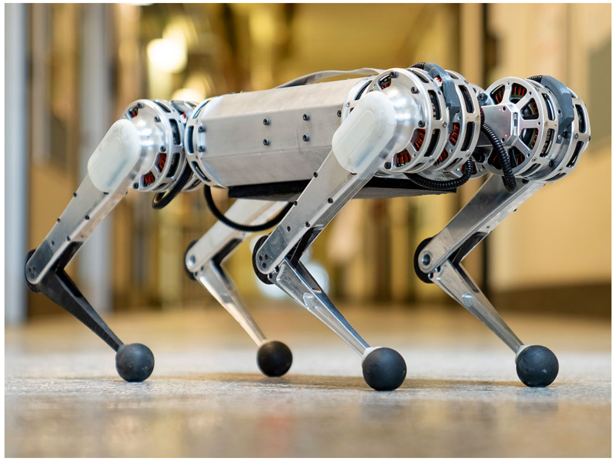


https://www.youtube.com/watch?v=M0ZXmGRCuts

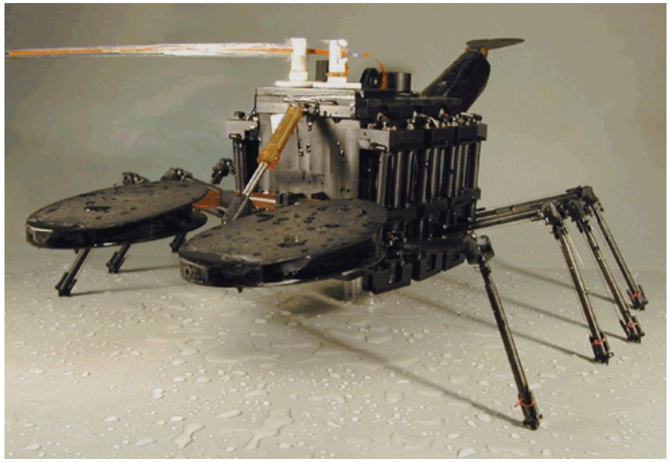
- Biped robots: compared to other multi-legged robots are more demanding regarding the dynamic balance
  - The balance is enhanced using a cooperation motion of the trunk and lower limbs
  - Are often developed with the idea to be used in daily life, interacting with human beings

https://www.youtube.com/watch?v=QdQL11uWWcI

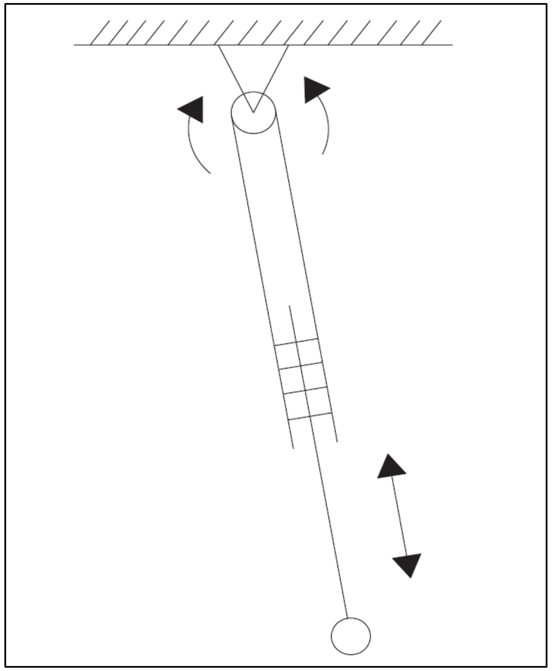
- Quadruped robots: can have both dynamic and static stability
  - Fast motion can be achieved easier than in bipedal case, given the possibility to keep at least two feet in contact during a dynamic gait
  - Legs' coordination is less complex than the case of robots with more legs



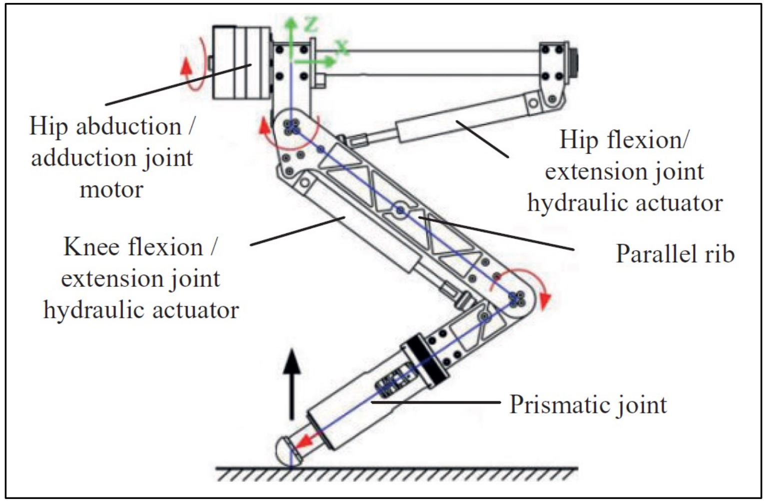
- Robots with six or more legs: the walking is always statically stable
  - It may be possible to recover the locomotion, though at reduced speed, from accidental situations when some legs are disabled
  - Control law for the coordination and control of the large number of leg joints becomes quite complex

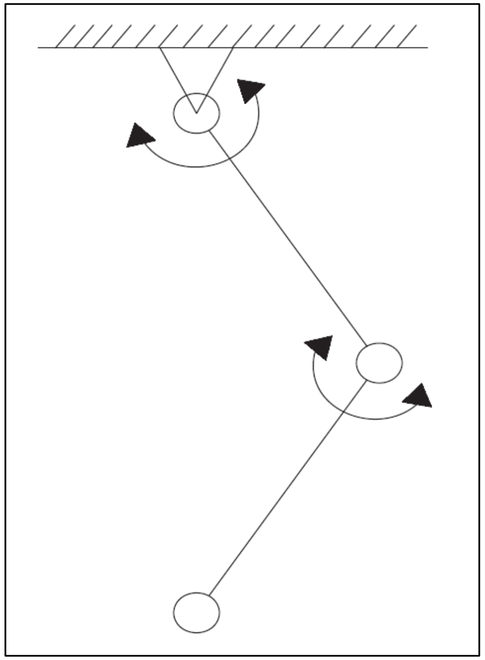


- Leg's configuration
  - Prismatic leg: it consists mainly of a rotating joint and a linearly moving prismatic joint
    - The structure is the simplest and lightest, with the lowest inertia
    - It limits its own kinematic performance due to fewer rotating joints, which results in insufficient terrain adaptability
    - Usually used in monopod robot, it's also been successfully used in quadruped and hexapod

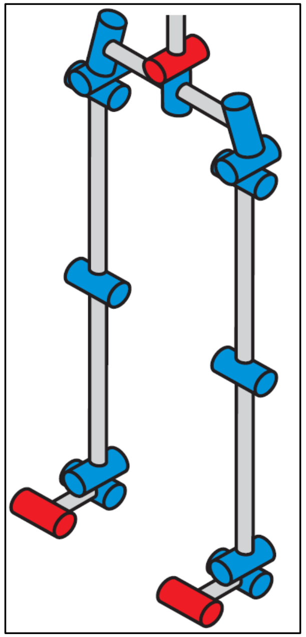


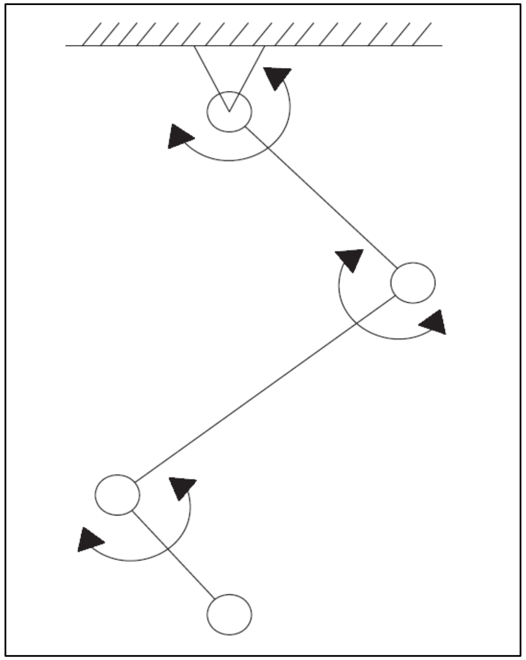
- Articulated leg: compared with prismatic legs, the articulated leg uses a rotating joint instead of a linear prismatic joint to achieve leg length control
  - Usually used in robot with four and more legs, with the addition of a perpendicular joint that enables the lateral motion



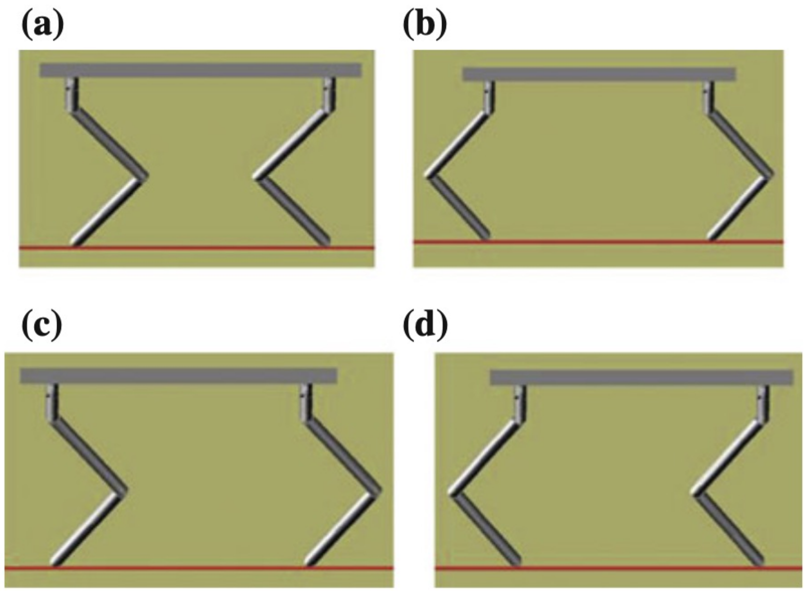


- Redundant articulated legs: the extension of the articulated legs with at least one more rotating joint
  - Usually used in biped robots, with the addition of other joints along the different axes that are useful to improve the stability



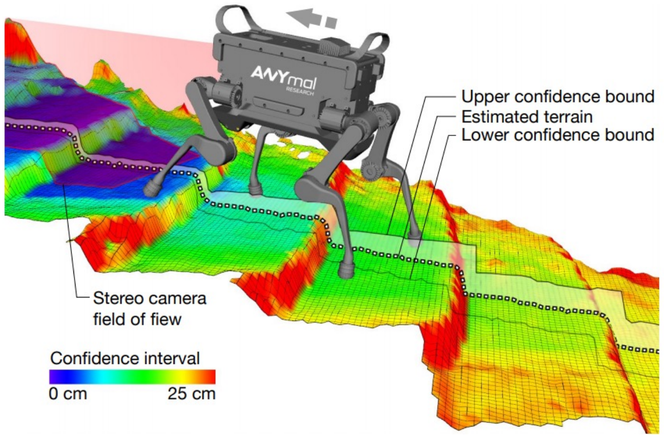


- Quadruped taxonomy
  - A) Forward elbow, backward knee
    - Better locomotion stability
    - Little impact force between the foot and the ground and good stability performance
    - Optimal configuration
  - B) Forward knee, backward elbow
  - C) Knees
    - Good locomotion stability
  - D) Elbows
    - Hind leg dragging problem in high-speed dynamic walking not compensated (less walking distance than ideal case)
    - Push-off impulse facilitated by the extension of the knee joint

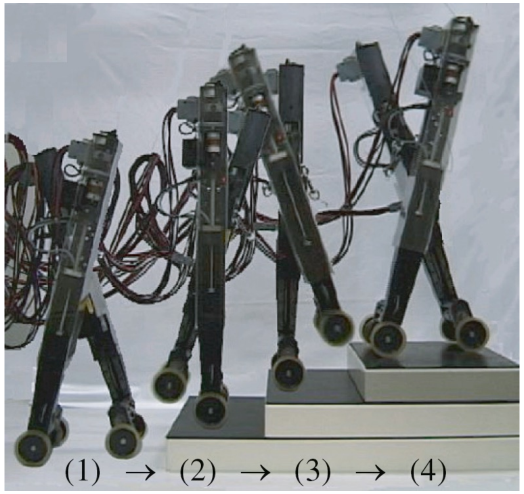


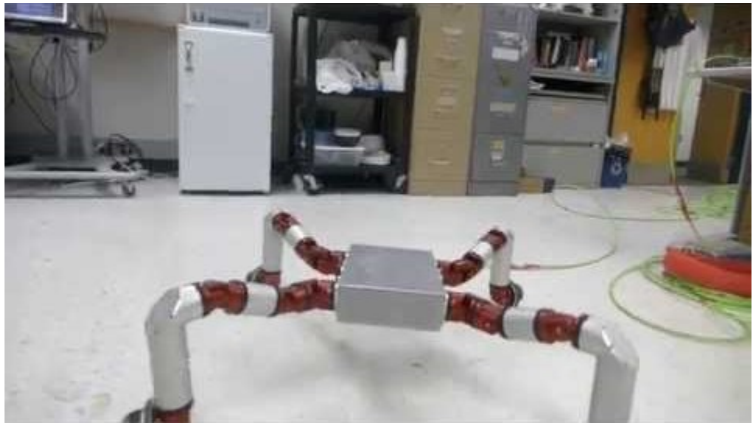
- Major challenge: designing actuator systems for high dynamic legged robots
  - Need to maximize torque, bandwidth, and power
  - Need to minimize friction, inertia, and mass loss
  - New actuators designs to reduce the effects of the high mechanical impedance resulting from the impact with the ground
    - Series Elastic Actuators (SEAs): minimize mechanical impedance by connecting the spring element in series with a high impedance actuator
    - Proprioceptive actuators (PA) or Quasi-Direct Drive Actuators (QDD): uses a large radius stator and rotor, minimizes mechanical inertia and gear ratio to make the system light when subjected to impact
    - The inertia of Proprioceptive Actuators is significantly much lower than other type of actuators and the response bandwidth is wider
  - Hydraulic actuator is another type of actuator commonly used
    - Higher output power, power density and bandwidth, faster response, and stronger anti-interference ability compared with electromagnetic actuators

- Sensors used
  - 6-axis Foot Area Sensor
    - Measure of the centre of pressure position
  - Magnetic Compass
    - Absolute orientation of the robot using the earth's magnetic field
  - Gyroscope
    - Used for the stabilization
  - Inertial Measurement Unit (IMU)
    - Linear accelerations and angular velocities
    - Gives the orientation of the robot with respect to the gravity field
  - Contact sensors at the feet
    - Detect the impact with the ground and measure the ground reaction forces
  - Joint encoders
    - Used to compute position and velocity of the feet
  - Obstacle detection and map reconstruction
    - LIDAR
    - Stereo camera
    - RGB-D sensor



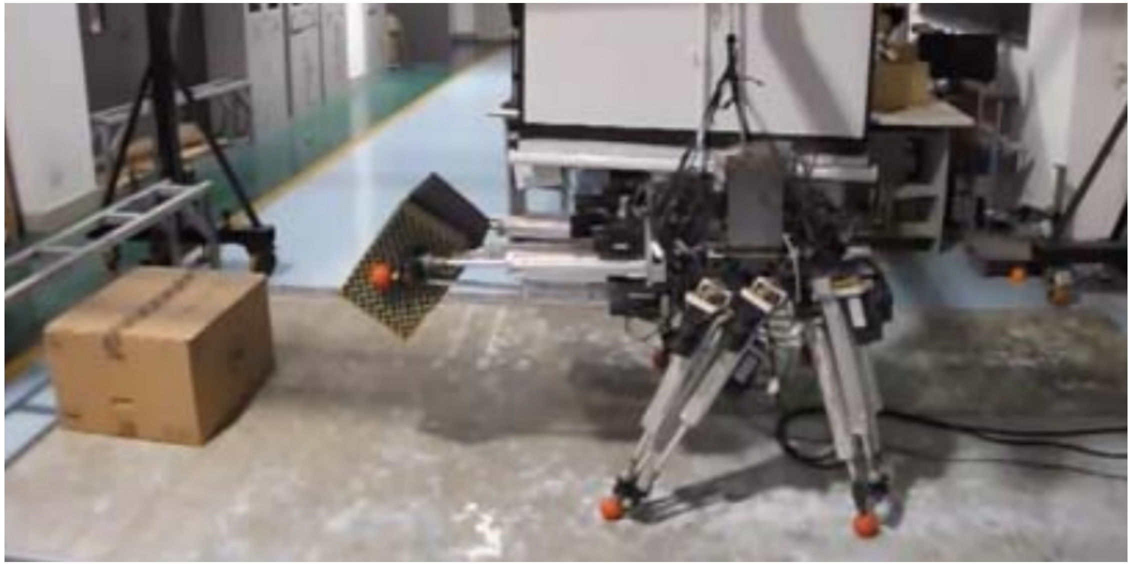
- There exists some hybrid legged robots
  - Leg-Wheel Hybrid Robots: a wheel is attached at the end of each leg
    - Can have both a legged gait and a wheeled locomotion
    - Legged locomotion guarantees high adaptability for rough terrain
    - Wheeled locomotion guarantees high speed





https://www.youtube.com/watch?v=Vhgft53qfDQ

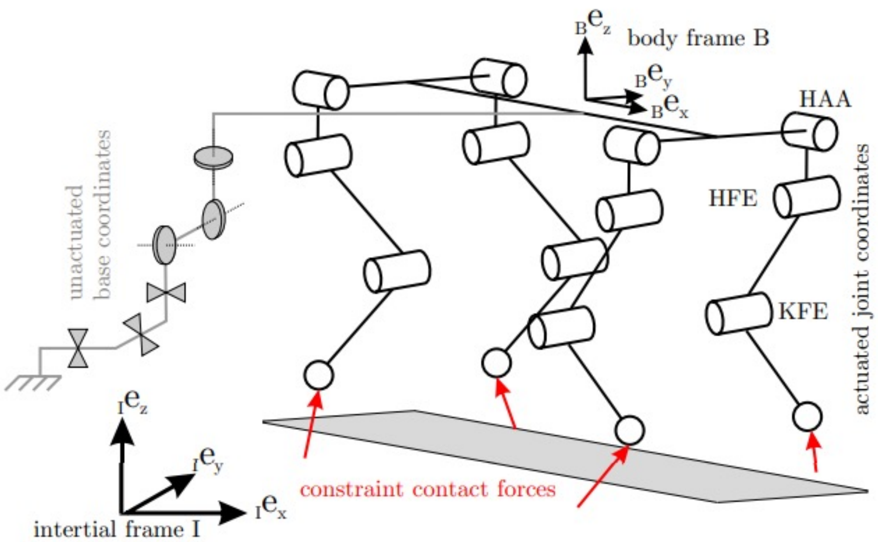
- Leg-Arm Hybrid Robots : can use legs as manipulators
  - An hexapod can manipulate an object by two legs while standing with four other legs



- Legged robots with arms
  - Mobile manipulators

https://www.youtube.com/watch?v=wXxrmussq4E

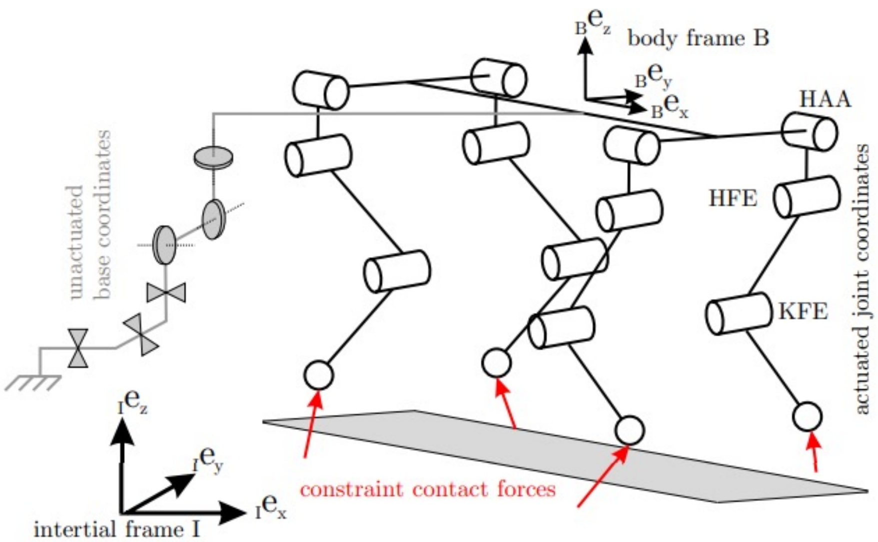
- The free-floating base can be described by
  - $p_b \in \mathbb{R}^3$
  - $R_b \in SO(3)$
- The joints of the robots are collected in the vector
  - $q_j = [q_1 \quad \cdots \quad q_{n_j}]^T$
- The dimension of the generalized coordinates vector $q = \begin{pmatrix} q_b \\ q_j \end{pmatrix} \in \mathbb{R}^{n_b+n_j}$ depends on the parameterization of the rotation
  - $q_b = \begin{bmatrix} p_b \\ \eta_b \end{bmatrix} \in \mathbb{R}^{n_b}$
  - The minimal number of coordinates for the base is $n_b = 6$



- Since differentiation in SO(3) is different from $\mathbb{R}^3$, the generalized velocity and acceleration vectors can be introduced

$$v = \begin{pmatrix} \dot{p}_b \\ \omega_b \\ \dot{q}_1 \\ \vdots \\ \dot{q}_{n_j} \end{pmatrix} \in \mathbb{R}^{6+n_j} \qquad \dot{v} = \begin{pmatrix} \ddot{p}_b \\ \dot{\omega}_b \\ \ddot{q}_1 \\ \vdots \\ \ddot{q}_{n_j} \end{pmatrix} \in \mathbb{R}^{6+n_j}$$

  - ω_b the angular velocity of the base



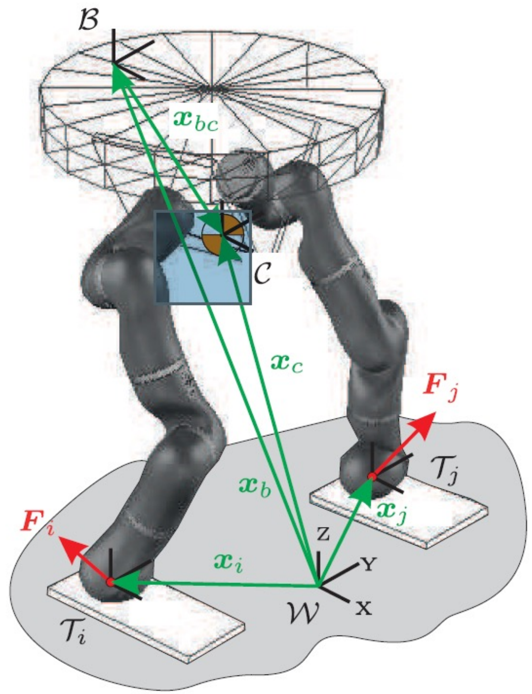
Stance and swing feet

- In a legged robot, contacts between each stance foot and the environment can be modelled as kinematic constraints
- Every foot T_i that is in contact with the environment imposes three constraints:

1. $p_i = const$
2. $\dot{p}_i = 0_3$
3. $\ddot{p}_i = 0_3$

  - $\mathrm{p_i} \in \mathbb{R}^3$, $\dot{\mathrm{p}}_\mathrm{i} \in \mathbb{R}^3$ and $p_\mathrm{i} \in \mathbb{R}^3$ position, velocity, and acceleration of the foot, respectively.

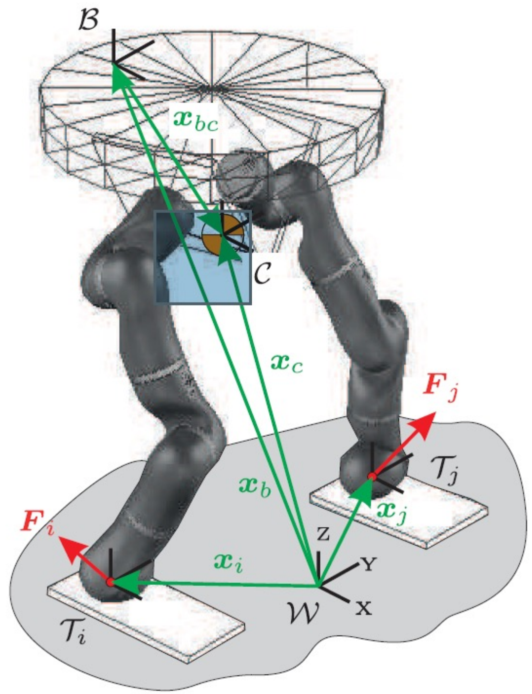
- The swing feet are the ones not in contact with the ground



- The contact constraints can be expressed as a function of the generalized velocities and accelerations using the contact point Jacobian
- The contact point Jacobian, mapping the generalized velocity vector into the generalized velocity vector of the *i*-th stance foot is

$$J_{T_i} = \begin{bmatrix} I_3 & S(p_{bi}) & J_i \\ O_{3\times 3} & I_3 & \end{bmatrix} \in \mathbb{R}^{6\times(n_b+n_j)}$$

  - with $p_{bi} = p_i - p_b$ and $J_i = \frac{\partial(r_i - q_b)}{\partial q_j} \in \mathbb{R}^{6\times n_j}$, where $r_i \in \mathbb{R}^6$ is the vector containing the pose of the *i*-th stance foot, is the Jacobian mapping the joint velocities into the generalized velocity vector of the *i*-th stance foot

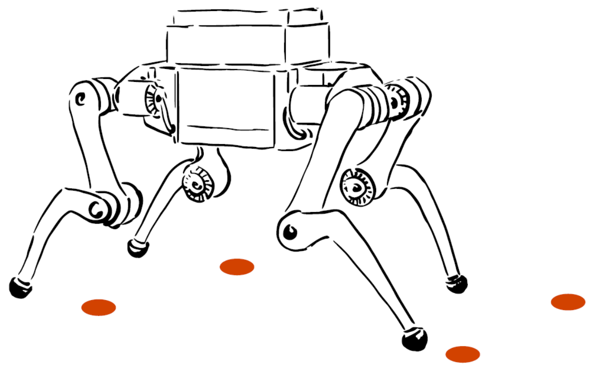


- For a point-foot robot (*i.e.*, a quadruped robot) the pose of the foot coincides with the position $p_i \in \mathbb{R}^3$
  - Then, the contact point Jacobian becomes

$$J_{T_i} = \begin{bmatrix} I_3 & S(p_{bi}) & J_{p,i} \end{bmatrix} \in \mathbb{R}^{3\times(n_b+n_j)}$$

  - $J_{p,i}$ considers the first three rows of J_i
- In case there are n_{st} stance feet, considering a point foot robot, the constraints are stacked into the following contact Jacobian

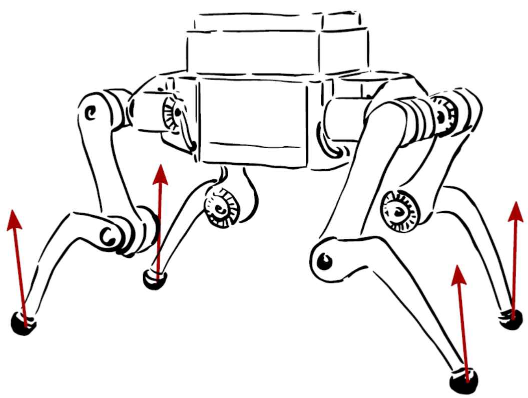
$$J_{st} = \begin{bmatrix} J_{T_1} \\ J_{T_2} \\ \vdots \\ J_{T_{n_{st}}} \end{bmatrix} = \begin{bmatrix} J_{st,b} & J_{st,j} \end{bmatrix} \in \mathbb{R}^{3n_{st}\times(n_b+n_j)}$$



  - where $J_{st,b}$ indicates the relation between the base motion and contact constraints

- Stance feet are in contact with the ground, applying a force to it. Vice versa, the ground applies a reaction force to the feet, defined as the *ground reaction force* (GRF).
  - When there is no contact, there is also no force

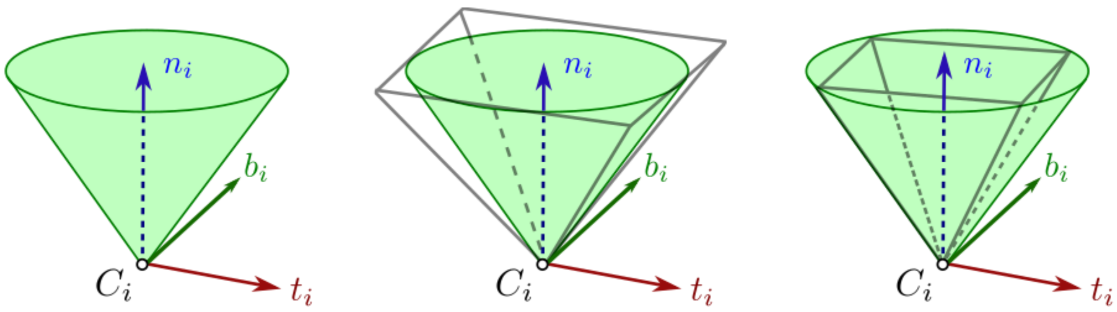
- Assuming rigid body behaviour for both the feet and the ground, the robot can only be pushed by the ground, not pulled by it
  - The GRF is also necessary in considering other constraints like foot slipping



- The GRF is also needed in the evaluation of foot slipping, and in particular in the computation of the friction between the foot and the ground
- In order for the foot to stick and not slip on the ground on the *i-th* contact point, the GRF must satisfy a certain condition, which depends on the *friction model* assumed
- The most used one is the Coulomb friction model, which is expressed in the form

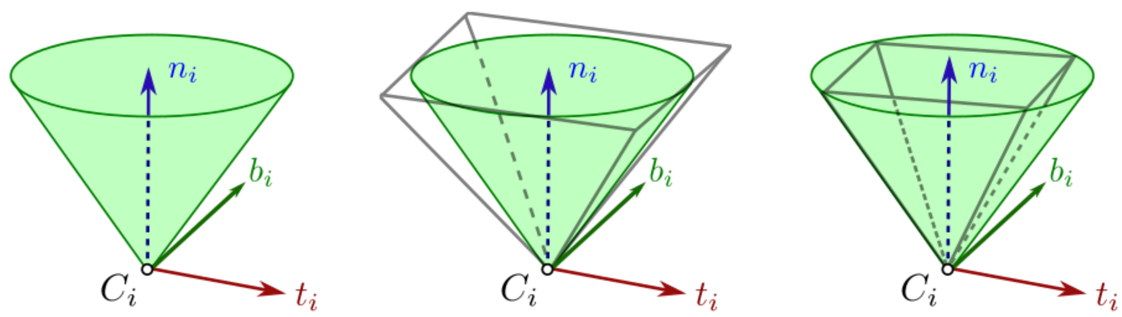
$$\sqrt{f_x^2 + f_y^2} = \mu f_z$$

where with f_x, f_y, f_z we are considering the projections of the GRF on the respective axes and with μ the friction coefficient



- This constraint is known as *Coulomb friction cone*, and is nonlinear. In order to ease the computation of such constraint, an alternative model has been developed as a simplification: the *friction pyramid.*
- With the friction pyramid, the constraint is split into two linear constraint, one per axis, giving origin to a square pyramid, but it becomes either more lax or more strict, considering either the circumscribed or the inscribed pyramid. The stricter version is more safe, and is described by the equations

$$|f_x| \leq \frac{\sqrt{2}}{2}\mu f_z$$
$$|f_y| \leq \frac{\sqrt{2}}{2}\mu f_z$$



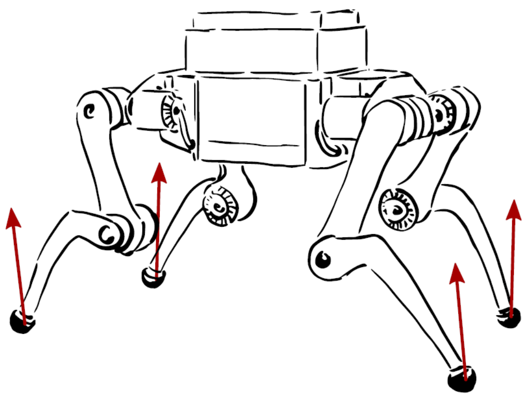
- The dynamic model of a floating base robot can be written as

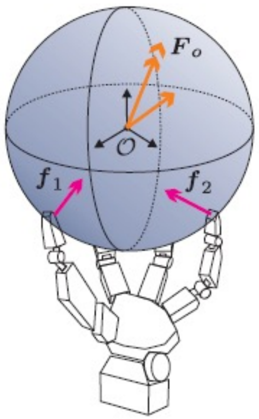
$$M(q)\dot{v} + Cv + g = S^T\tau + J_{st}^T f_{gr} + J^T f_{ext}$$

with :

- $M(q) \in \mathbb{R}^{(n_b+n_j)\times(n_b+n_j)}$ the inertia matrix
- $C \in \mathbb{R}^{(n_b+n_j)\times(n_b+n_j)}$ accounts for Coriolis and centrifugal forces
- $g \in \mathbb{R}^{(n_b+n_j)}$ accounts for the gravitational force
- $S = \begin{bmatrix} O_{n_j\times 6} & I_{n_j} \end{bmatrix}$ is the torque distribution matrix, selecting the actuated joints
- $f_{gr} \in \mathbb{R}^{3n_{st}}$ the ground reaction forces
- $\tau \in \mathbb{R}^{n_j}$ the joint actuation torques
- J the stacked matrix of the feet Jacobian
- f_{ext} the stacked vector containing the net force at the legs' tips accounting for unmodelled dynamics and disturbances at any point of the robot

- This dynamic model represents the robot as a single rigid body subject to forces at the contact patches
- The base follows the desired trajectory through the modulation of ground reaction forces, while retaining the balance
- The problems of balancing a robot and grasping an object are similar, in the sense that both try to achieve a desired wrench (on the robot or on the object) based on the application of suitable forces at the contact points (at the feet or at the fingertips, respectively).





Constrained consistent dynamics

- As said, contacts are usually handled as kinematic constraints
  - For the *i*-th stance foot the constraint is

$$J_{T_i}\dot{v} + \dot{J}_{T_i}v = 0$$

- A legged robot is a constrained system when there is a contact with environment
  - The legged robot can move ONLY thanks the contact with the environment
  - The control is possible thanks to the contact with the ground: the contact should be exploited in this case: controlling contacts mean that the robot can perform physical work to the environment, that is the point of robotics, after all (!)
- To apply the inverse dynamics, a constrained consistent dynamics can be retrieved

- The constrained consistent dynamics can be obtained using a dynamically consistent support null-space matrix as

$$N_c = I_{n_{st}} - J_{st}^{\#} J_{st}$$

  - $J_{st}^{\#} = M^{-1} J_{st}^T (J_{st} M^{-1} J_{st}^T)^{-1}$ is the dynamically consistent pseudo-inverse of the contact Jacobian
  - $N_c \in \mathbb{R}^{(n_b+n_j)\times(n_b+n_j)}$ defines a generalized space of motion with no acceleration or force coupling effects on the stance legs
  - N_c is an orthogonal projection operator, such that $N_c J_{st}^T = 0$ and $N_c = {N_c}^2 = {N_c}^T$
- The constrained consistent dynamic is

$$N_c(M\dot{v} + Cv + g - J^T f_{ext}) = N_c S^T \tau$$

  - where we are getting rid of the ground reaction forces
- Given a desired acceleration $\dot{v}_d$, the constraint consistent equation of motion can be inverted to obtain the desired joint torques

$$\tau^* = (N_c S^T)^{\#} N_c (M\dot{v}_d + Cv + g - J^T f_{ext})$$

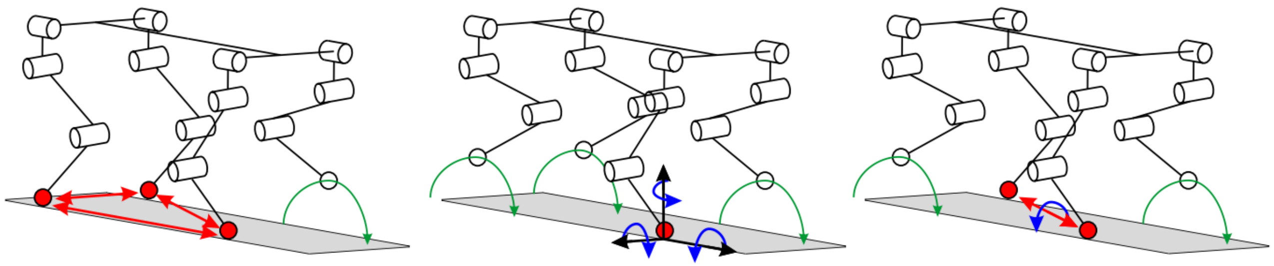
Constrained consistent dynamics

- There exists a null-space that allows to modify τ^*

$$\tau^* = (N_c S^T)^\# N_c (M\dot{v}_d + Cv + g) + \mathcal{N}(N_c S^T)\tau_0^*$$

- There exist different joint torques distributions which all lead to the same desired motion of the system
  - The different torque distributions change the contact force distribution
  - In case of multiple contacts, internal forces can be created between the contacts which does not change the net force and moment on the robot and thus does not create any additional accelerations

- In case $rank(J_{st,b}) = 6$, the robot is fully constrained and the base motion (we focus only on this now) can be controlled through joint motion
- For a quadruped robot:
  - with a three-point contact $rank(J_{st,b}) = 6$, the body position and orientation is fully controllable through the joints
  - with a two-point contact $rank(J_{st,b}) = 5$, the system is underactuated and the base can not be arbitrarily moved by the joints (the robot cannot change the orientation around the line of support)



(a) Fully-constraints (b) Under-constrained

- Similar reasonings can be applied to J_{st} and the constrained consistent dynamics for the whole configuration

- Since the location of the center of mass (CoM) is crucial for balancing, a coordinate transformation can be performed to replace p_b with the CoM location
  - Notice that, before, we considered the position of the base that, in general, is not coincident withe the center of mass
- Defining a frame C located at the CoM, with the position $p_c \in \mathbb{R}^3$and with the same orientation R_b as the base link, the generalized coordinate vector becomes

$$q_c = \begin{bmatrix} p_c \\ \eta_c \\ q_j \end{bmatrix}$$

- The relation between the base velocity and the CoM velocity can be written as

$$\begin{pmatrix} \dot{p}_c \\ \omega_c \\ \dot{q}_j \end{pmatrix} = \underbrace{\begin{bmatrix} I_3 & -S(p_{bc}) & J_{bc} \\ 0 & I_3 & 0 \\ 0 & 0 & I_{n_j} \end{bmatrix}}_{\bar{T}} \begin{pmatrix} \dot{p}_b \\ \omega_b \\ \dot{q}_j \end{pmatrix}$$

  - with
    - $p_{bc} = p_c - p_b$
    - $J_{bc} = \frac{\partial p_{bc}}{\partial q_j}$

Centroidal dynamics derivation

- Using the transformation matrix $\bar{T}$, the dynamic model becomes

$$\underbrace{\begin{bmatrix} mI_3 & O_{3\times3} & O_{3\times n_j} \\ O_{3\times3} & M_{11} & O_{3\times n_j} \\ O_{n_j\times3} & O_{n_j\times3} & M_{22} \end{bmatrix}}_{\bar{M}} \dot{v}_c + \underbrace{\begin{pmatrix} 0_3 \\ \bar{C}_1 \\ \bar{C}_2 \end{pmatrix}}_{\bar{C}} v_c + \begin{pmatrix} mg_0 \\ 0_3 \\ 0_{n_j} \end{pmatrix} = S^T\tau + \underbrace{\begin{pmatrix} \bar{J}_{st,c}^T \\ \bar{J}_{st,j}^T \end{pmatrix}}_{\bar{J}_{st}^T} f_{gr} + \underbrace{\begin{pmatrix} \bar{J}_c^T \\ \bar{J}_j^T \end{pmatrix}}_{\bar{J}^T} f_{ext}$$

- where
  - m is the mass of the robot
  - $\bar{M} = \bar{T}^{-T} M\, \bar{T}^{-1}$
  - $\bar{C} = \bar{T}^{-T} C\, \bar{T}^{-1} + \bar{T}^{-T} M \frac{d\bar{T}^{-1}}{dt}$
  - $\bar{J}_{st} = J_{st}\bar{T}^{-1}$
  - $\bar{J} = J\bar{T}^{-1}$
  - g_0 is the gravity acceleration
  - $v_c = \begin{pmatrix} \dot{p}_c \\ \omega_c \\ \dot{q}_j \end{pmatrix}$

Further simplifications

- If the assumptions that the main body's angular motion is slow, and that the legs' mass is negligible with respect to the robot's total mass are made, then the term $\bar{C}_1$ can be neglected
- The obtained dynamics decouple the centroidal dynamics (related to the CoM) from the dynamics of the legs
- If the external disturbances are compensated for, the first three rows of the dynamics correspond to Newton's law describing the CoM motion of the overall system

$$m\ddot{p}_c = mg_0 + \sum_{i=1}^{n_{st}} f_{gr,i}$$

  - $f_{gr,i}$ is the ground reaction force at the *i*-th foot
  - This equation makes it obvious that the robot needs ground reaction forces in order to move its COM in a direction other than that of gravity

- The rows from 4 to 6 contain the equations of motion for the hip rotation

$$I\dot{\omega} = \sum_{i=1}^{n_{st}} p_{ci} \times f_{gr,i}$$

  - p_{ci} is the vector from the center of mass (COM) to the point where the force of the *i*-th foot is applied
    - $p_{ci} = p_i - p_c$
- Taking in account the centroidal dynamics, the kinematic constraint for the *i*-th stance foot is

$$\bar{J}_{T_i}\dot{v}_c + \dot{\bar{J}}_{T_i} v_c = 0$$

  - with $\bar{J}_{T_i} = J_{T_i}\bar{T}^{-1}$
  - For a point-foot robot, it assumes the form

$$J_{T_i} = [I_3 \ \ S(p_{ci}) \ \ \bar{J}_i] \in \mathbb{R}^{3\times(n_b+n_j)}$$

- Given the decoupled structure of the dynamic model after the coordinate transformation to the CoM, if the external disturbances are compensated for, the inverse dynamics can be obtained from the last rows of the model, representing the joints dynamics

$$\tau^* = M_{22}\ddot{q}_j^* + \bar{C}_2 \upsilon - \bar{J}_{st,j} f_{gr}^*$$

  - $\ddot{q}_j^*$ is the desired joints acceleration vector
  - f_{gr}^* are the desired ground reaction forces

- When dealing with contacts with the ground some key assumptions must be made, and the model must be adapted.

- Considering no contact with the ground, the dynamic model can be rewritten as

$$\bar{M}(q_c)\dot{v}_c + \bar{C}(q_c, v_c)v_c + \bar{g}(q_c) = S^T\tau$$

where $\bar{g}$ is the gravity contribution term

- To integrate also ground contacts, the complementarity constraints must be added, and the ground reaction forces must be kept into consideration. This leads to the definition of the *complementarity dynamic system*

$$\begin{cases} \bar{M}(q_c)\dot{\bar{v}}_c + \bar{C}(q_c, v_c)v_c + \bar{g}(q_c) = S^T\tau + \nabla F(q_c)\lambda_n + P_t(q_c, v_c) \\ \lambda_n^T F(q_c) = 0 \\ \lambda_n \geq 0 \\ F(q_c) \geq 0 \end{cases}$$

- Note that:
  - With $F(q_c)$ we are considering the distance of the *m possible* contact points on the feet from the ground
  - With λ_n we are considering the normal of the GRF in correspondence of these points
  - With $P_t(q_c, v_c)$ we are considering the tangential components of the GRF

Complementarity Constraints

- The complementarity constraints are added to account for
  - The ground can only push on the robot, not pull
  - The feet can only impact the ground, not penetrate it
  - If the distance between the *i-th* point and the ground is greater than zero, then no reaction force can be present
  - If there is a reaction force on the *i-th* point, then its distance from the ground must be zero
- These considerations must be kept in mind while developing a controller
- Notice that the loss of contact may result in a discontinuity of the reaction force, and vice versa, an impact must be treated carefully. Impacts are generally modeled via the introduction of restitution forces.

- Recall the centroidal dynamics

$$m\ddot{p}_c = mg_0 + \sum_{i=1}^{n_{st}} f_{gr,i}$$

$$\dot{L} = I\dot{\omega} = \sum_{i=1}^{n_{st}} p_{ci} \times f_{gr,i}$$

  - L is the angular momentum
- Without loss of generality, suppose that all the contact points of the stance legs are with the third component equal to zero
  - $p_i^z = 0$
- Consider the sum of the angular part with the cross product of p_c with the linear part

$$mp_c \times (\ddot{p}_c - g_0) + \dot{L} = \sum_{i=1}^{n_{st}} p_i \times f_{gr,i}$$

  - Notice the sum on the right side has p_i instead of p_{ci}

- Divide the previous equation by the third equation of the linear part

$$\frac{m p_c \times (\ddot{p}_c - g_0) + \dot{L}}{m(\ddot{p}_c^z - g_0^z)} = \frac{\sum_{i=1}^{n_{st}} p_i \times f_{gr,i}}{\sum_{i=1}^{n_{st}} f_{gr,i}^z}$$

- Since $p_i^z = 0$, then the x and y coordinates can be written as

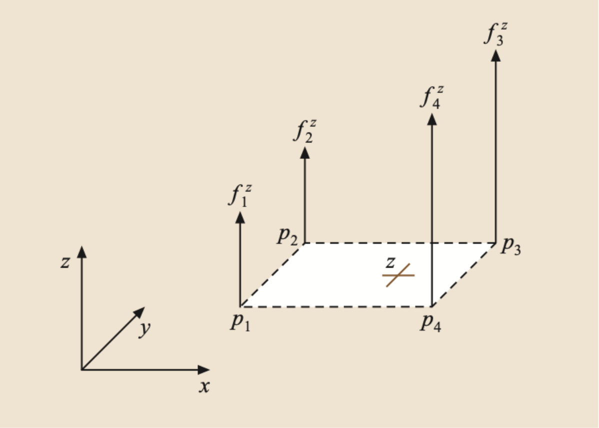
$$p_c^{x,y} - \frac{p_c^z}{\ddot{p}_c^z - g_0^z}(\ddot{p}_c^{x,y} - g_0^{x,y}) + \frac{1}{m(\ddot{p}_c^z - g_0^z)} S\dot{L}^{x,y} = \boxed{\frac{\sum_{i=1}^{n_{st}} f_{gr,i}^z p_i^{x,y}}{\sum_{i=1}^{n_{st}} f_{gr,i}^z}}$$

  - with $S = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \in SO(2)$

Center of pressure (CoP), denoted as $p_z \in \mathbb{R}^3$, of the contact forces $f_{gr,i}$

- Since the ground reaction forces are unilateral (the ground can push the robot, not pull)
  - $f_{gr,i}^{z} \geq 0$
- Then, the CoP is bound to lie in the convex hull of the contact points
  - $p_z^{x,y} = \frac{\sum_{i=1}^{n_{st}} f_{gr,i}^{z} p_i^{x,y}}{\sum_{i=1}^{n_{st}} f_{gr,i}^{z}} \in conv\{p_i^{x,y}\}$
- Therefore, this reveals an ordinary differential inclusion (ODI)

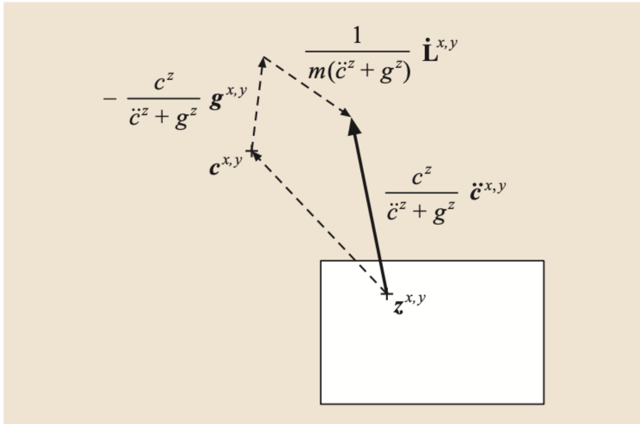
$$p_c^{x,y} - \frac{p_c^z}{\ddot{p}_c^z - g_0^z}(\ddot{p}_c^{x,y} - g_0^{x,y}) + \frac{1}{m(\ddot{p}_c^z - g_0^z)} S\dot{L}^{x,y} = p_z^{x,y} \in conv\{p_i^{x,y}\}$$



- Such a ODI bounds the motion of the robot's CoM and the variation of its angular momentum with respect to the position of the contact points of the stance legs
- It is possible to rearrange the ODI as follows to highlight its geometrical meaning

$$\frac{p_c^z}{\ddot{p}_c^z - g_0^z}(\ddot{p}_c^{x,y} - g_0^{x,y}) = (p_c^{x,y} - p_z^{x,y}) + \frac{1}{m(\ddot{p}_c^z - g_0^z)} S\dot{L}^{x,y}$$

  - Aside from the effects of gravity and variations of the angular momentum, the horizontal acceleration of the CoM is the result of a force pushing the CoM away from the CoP, which is bound to lie in the convex hull of the contact points
  - This is an intrinsically unstable dynamics



- From the definition of CoP it is possible to write

$$p_z^{x,y} = \frac{\sum_{i=1}^{n_{st}} f_{gr,i}^{z} p_i^{x,y}}{\sum_{i=1}^{n_{st}} f_{gr,i}^{z}} \Rightarrow \sum_{i=1}^{n_{st}} f_{gr,i}^{z} \left(p_z^{x,y} - p_i^{x,y}\right) = 0$$

  - It is possible to recognize that the definition of CoP implies that the horizontal momenta of the ground reaction forces with respect to the CoP are zero
  - The CoP is then also referred to as zero moment point (ZMP)

- The basic question to answer is often "*Will the robot fall down?*"
- There are several useful concepts relating to stability referred to the previous question, including
  - Fixed points
    - Stable fixed points represent the static postures in which the robot can safety stand still
  - Limit cycles
    - They provide a natural extension of the fixed points analysis to periodic walking or running motions
  - Viability
    - It is a concept of controlled invariance, which analyses the set of states from which the robot is able to avoid to fall
  - Controllability
    - It provides a slightly restricted notion of viability, analysing the set of states from which the robot is capable of returning to a particular fixed point (or limit cycle)

- In addition, if there are errors in the robot model, uncertainty about the environment, unmodeled disturbances, and so on, robustness and stochastic theories may be used
  - Robust stability
    - It examines the properties of the system considering worst-case (bounded) disturbances
  - Stochastic stability
    - Stochastic analysis provides tools to investigate the probability of falling down
  - Input-output stability
    - This analysis treats a particular disturbance as an input, a performance criteria as output, and attempts to compute a relative gain or sensitivity of the robot performance due to this input
  - Stability margins
    - In practice control designers often settle for the system staying comfortably away from the boundaries of deterministic stability

- Fixed points
  - A fixed point of the system is a posture in which the robot stand still
  - In static conditions this happens when $\ddot{p}_c = \dot{L} = 0$
  - The ODI is then modified as follows

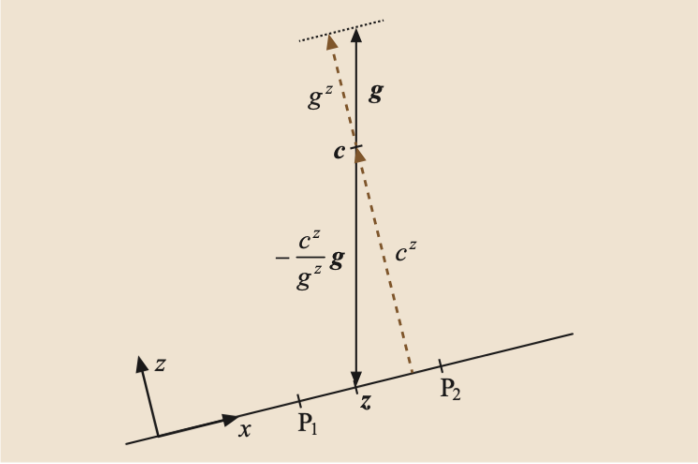
$$p_c^{x,y} - \frac{p_c^z}{g_0^z} g_0^{x,y} = p_z^{x,y} \in conv\{p_i^{x,y}\}$$

  - This necessary condition states that the CoM must project on the ground along the gravity vector inside the convex hull of the contact points of the stance legs
    - This convex hull can be now defined as the support polygon

- ## Fixed points
  - A fixed point of the system is a posture in which the robot stand still
  - In static conditions this happens when $\ddot{p}_c = \dot{L} = 0$
  - The ODI is then modified as follows

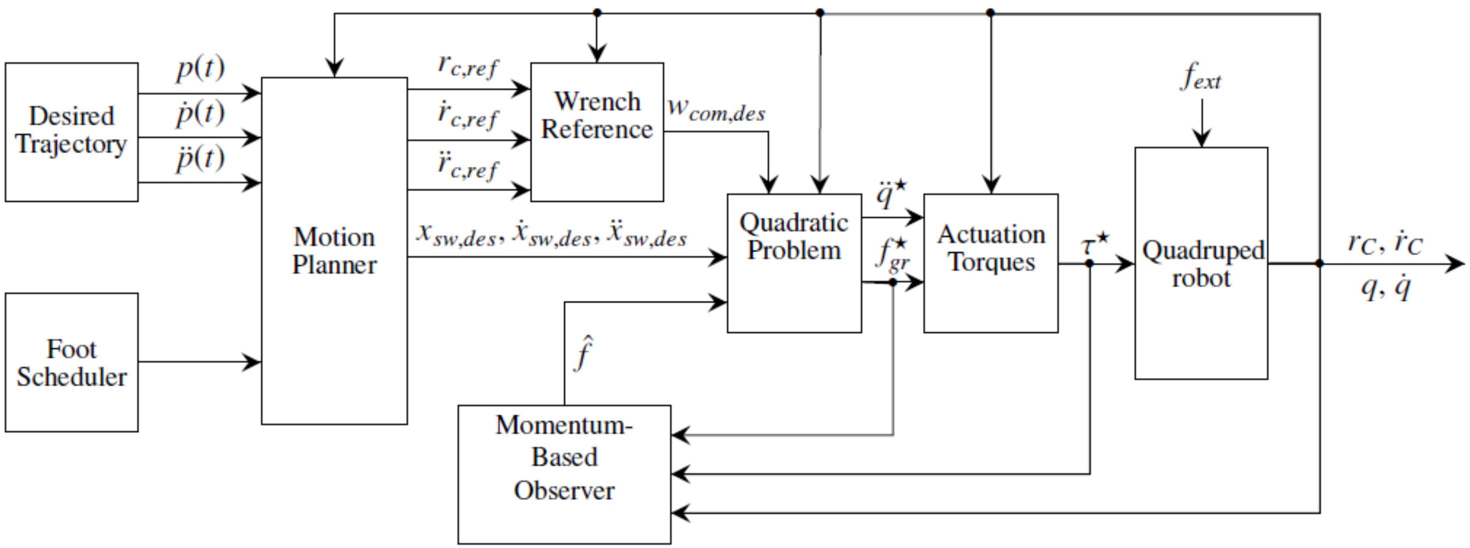
$$p_c^{x,y} - \frac{p_c^z}{g_0^z} g_0^{x,y} = p_z^{x,y} \in conv\{p_i^{x,y}\}$$

  - This necessary condition states that the CoM must project on the ground along the gravity vector inside the convex hull of the contact points of the stance legs
    - This convex hull can be now defined as the support polygon
  - Once a fixed point has been found, one would often like to examine its stability
    - Local stability can be established by linearizing the robot dynamics at the fixed point
    - Once the local stability is established, it may also be possible to understand the region of attraction of the fixed point

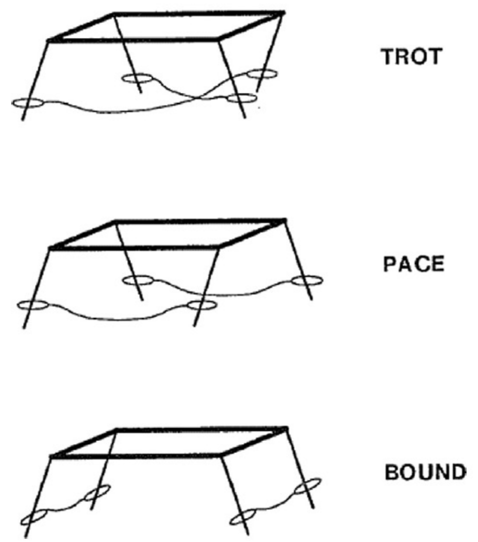


- Limit cycles
  - A natural extension of fixed-point analysis to walking and running motions is to examine the existence and stability of periodic orbits, or limit cycles, of the dynamical system
  - Robots with more actuators typically have an abundance of periodic solutions and planning techniques can be used to find solutions that, for instance, minimize a quantity of interest such as the cost of transport or a measure of open-loop stability
  - Under mild conditions, a sufficient criteria for establishing orbital stability is establishing fixed-point stability of a Poincaré map
    - Due to the limitation that finding periodic solutions is typically done numerically, Poincaré map analysis is also typically done numerically (*e.g.*, using finite differences to evaluate a local linearization of the map)
    - Note that limit cycle analysis can be applied (carefully) to solutions that are not periodic in all states
      - For instance, the horizontal position of the floating base should be left out if the robot needs to make forward progress
  - However, useful locomotion through nontrivial environments will likely require aperiodic motions
    - Orbital stability of an (infinite) aperiodic trajectory can be established using a transverse linearization or moving Poincaré section

- The desired joints acceleration $\ddot{q}_j^*$ and the desired ground reaction forces f_{gr}^* can be obtained as the result of a whole-body controller
- A whole-body controller decouples the motion planning from the control



- Desired trajectory
  - A desired trajectory for the CoM is computed
- Foot Scheduler
  - The contact schedule is planned, based on the desired gait
  - For a quadruped robot different gait can be realized with two stance legs (highly dynamic gaits):
    - Trot: cross legs move
    - Pace: lateral legs work in coordination
    - Bound: the front and the rear legs work in pairs
- Motion planner
  - The motion is continuously replanned following the desired trajectory
  - The reference for the CoM is computed on the base of the ZMP-criterion, guaranteeing the balance of the robot
- Wrench reference
  - Based on the CoM reference, a desired wrench for the CoM is computed
  - This wrench must be generated through the application of the desired ground reaction forces obtained by the controller

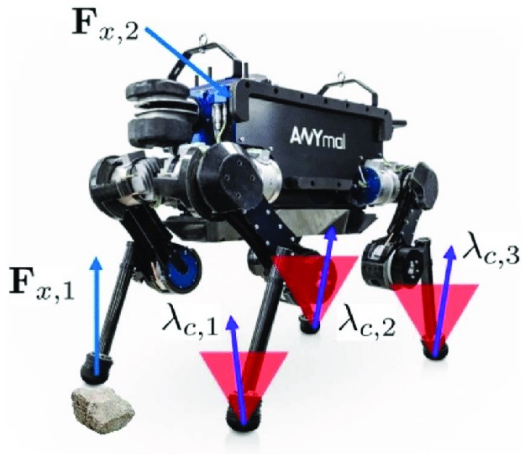


- Quadratic problem
  - The desired acceleration and the desired ground reaction forces are obtained throug a quadratic problem

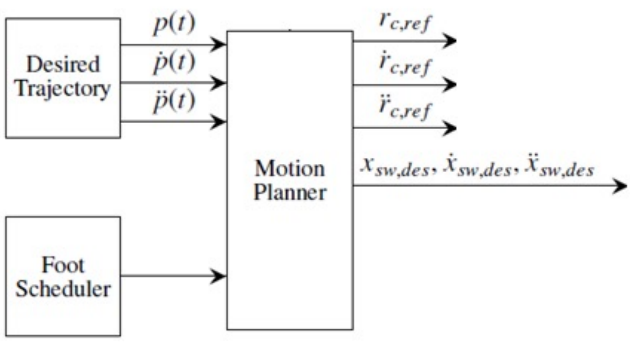
$$\begin{aligned} \underset{\zeta}{\text{minimize}} \quad & f(\zeta) \\ \text{subject to} \quad & A\zeta = b, \\ & D\zeta \leq c. \end{aligned}$$

  - The constraints regard
    - Dynamic consistency
    - Non-sliding contact
    - Torque limits
    - Swing legs task

- Actuation torques are computed through the inverse dynamics
- Momentum-based observer
  - External forces are computed to be compensated for inside the controller

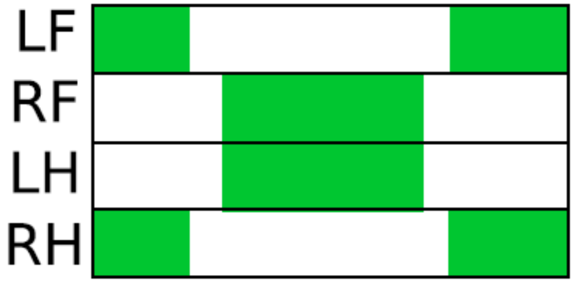


- The motion planner takes as input a desired trajectory for the CoM and a foot scheduling
- It gives as output the references for the CoM and the swing feet
- Desired trajectory
  - It can be considered as a regular path. Indeed, continuous update of the motion plan can cause drift in the motion of the torso with respect to the reference footholds. This can be caused by the accumulation of control errors, which alter the solution such that the motion can become unfeasible. To avoid this issue, a desired trajectory is used to adjust the movement of the robot during each replanning phase.



Motion planner

- The desired path is approximated as a sequence of splines represented by $p(t), \dot{p}(t)$, and $\ddot{p}(t)$. The regularized path is computed such that:
  - The initial state $p(0)$, $\dot{p}(0)$ and $\ddot{p}(0)$ coincides with the center of the initial support polygon and the initial velocity and acceleration
  - The final state $p(t_f)$, $\dot{p}(t_f)$ and $\ddot{p}(t_f)$ is set to be at the center the polygon defined by the planned footholds, which is the polygon that would support the robot if it would stop to stand at the end of the support polygon sequence
- Foot scheduler
  - The complete motion is divided into separate phases: stance phase and swing phase
  - For example, for quadrupedal trot, a standard sequence to follow is left front/right hind → right front/left hind



- Depending on the swing duration T_{sw} and the maximum step length the motion planner computes a sequence of footholds
- Stance phase
  - All the legs are in the stance condition, ensuring an intrinsic balance
  - The CoM reference $r_{c,ref}$, $\dot{r}_{c,ref}$, and $\ddot{r}_{c,ref}$ can be computed as a 3-rd order spline that tracks the desired trajectory
- Swing phase
  - At least one leg is swinging. In case of a quadruped robot, the motion could be quasi-static if only one leg is moving, or highly dynamic if two legs are swinging and the support polygon degenerates into a line
  - During this phase, both the references for the CoM and the swing feet need to be planned
  - The linear reference of the CoM is computed solving an optimization problem, having as variables the coefficients of a third-order spline for each coordinate of the CoM.
  - Considering the reference $r_{c,ref} = [\, p_{c,ref}^T \;\; \eta_{c,ref}^T]^T$, the motion of the coordinates of the CoM can be described by the *i*-th spline as

$$p_{c,ref,x}(t) = \alpha_{i3,x}t^3 + \alpha_{i2,x}t^2 + \alpha_{i1,x}t + \alpha_{i0,x}$$
$$p_{c,ref,y}(t) = \alpha_{i3,y}t^3 + \alpha_{i2,y}t^2 + \alpha_{i1,y}t + \alpha_{i0,y}$$
$$p_{c,ref,z}(t) = \alpha_{i3,z}t^3 + \alpha_{i2,z}t^2 + \alpha_{i1,z}t + \alpha_{i0,z}$$

- Considering the stacked vectors

$$\lambda(t) = [t^3 \;\; t^2 \;\; t \;\; 1]$$
$$\alpha_i = [\alpha_{i3} \;\; \alpha_{i2} \;\; \alpha_{i1} \;\; \alpha_{i0}]$$

  - where $\alpha_{ik} = [\alpha_{ik,x} \;\; \alpha_{ik,y} \;\; \alpha_{ik,z}]^T$.
- Defining

$$T = \begin{bmatrix} \lambda(t) & 0 & 0 \\ 0 & \lambda(t) & 0 \\ 0 & 0 & \lambda(t) \end{bmatrix}$$

the linear reference of the CoM for the *i*-th spline can be written as

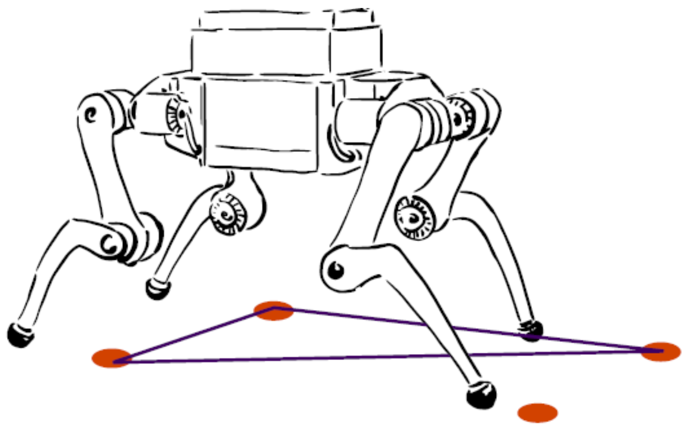
  - $p_{c,ref}(t) = T(t)\alpha_i$
  - $\dot{p}_{c,ref}(t) = \dot{T}(t)\alpha_i$
  - $\ddot{p}_{c,ref}(t) = \ddot{T}(t)\alpha_i$

- The motion planning algorithm is formulated as a nonlinear optimization problem which minimizes a generic cost function subject to equality and inequality constraints
- The variables of the problem are the coefficients of the spline stacked in the vector α_i
- The cost function minimizes the acceleration of the center of mass
- To ensure that each pair of adjacent splines is connected, junction constraints are set
  - The junction constraints for the *x* coordinate between the *k*-th and *(k + 1)*-th spline can be written as:

$$\begin{bmatrix} \lambda(t_{fk})^T & -\lambda(0)^T \\ \dot{\lambda}(t_{fk})^T & -\dot{\lambda}(0)^T \end{bmatrix} \begin{bmatrix} \alpha_k^x \\ \alpha_{k+1}^x \end{bmatrix} = 0$$

    - with t_{fk} representing the duration in seconds of spline s_k
  - Similarly, it can be written for the other coordinates

- Some inequality constraints need to be imposed to guarantee the balance, basing on a ZMP criterion
  - Balance during locomotion can be ensured by constraining ZMP to lie inside the support polygon
  - Starting from the measured feet configuration, the motion plan is called and, using the planned feet positions, a sequence of support polygons and their duration is computed by using the contact schedule
  - By always assuming at least two feet on the ground, the support polygon for a quadruped can be a line, a triangle, or a quadrilateral



- The location of the ZMP is a function of the motion of the center of mass p_c
- Recall the ODI

$$p_c^{x,y} - \frac{p_c^z}{\ddot{p}_c^z - g_0^z}(\ddot{p}_c^{x,y} - g_0^{x,y}) + \frac{1}{m(\ddot{p}_c^z - g_0^z)} S\dot{L}^{x,y} = p_z^{x,y}$$

- The location of the ZMP's *x*-coordinate is defined by

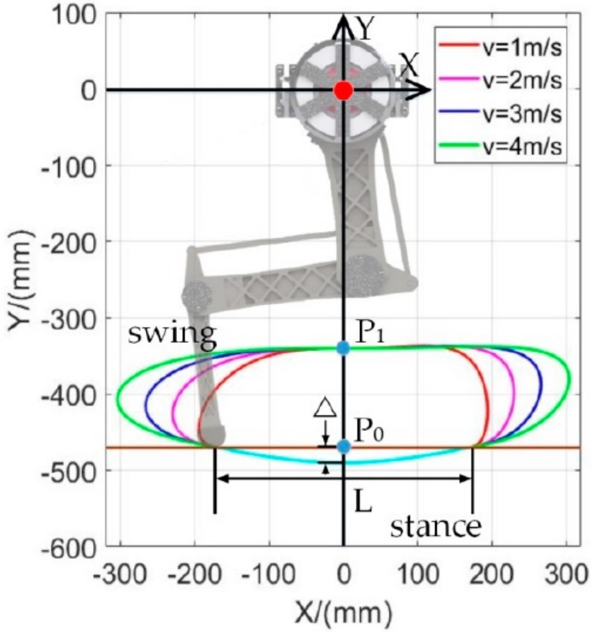
$$p_z^x = p_c^{x,y} - \frac{p_c^z}{\ddot{p}_c^z - g_0^z}(\ddot{p}_c^x - g_0^x) - \frac{1}{m(\ddot{p}_c^z - g_0^z)} \dot{L}^y$$

  - It can be assumed $\dot{L} = 0$ if there is no optimization for rotations within the trajectory computation
  - In case of flat ground, $g_0^z = -g$ and $g_0^x = 0$
- The *y*-coordinated can be defined similarly
- The ZMP can be limited inside the support polygon adding a constraint like

$$a p_z^x + b p_z^y + c \geq 0$$

  - where a, b, and c are the coefficients of the line that passes through the vertices of an edge of the support polygon
  - The normal vector to this line $n = [\,a \quad b\,]^T$ is directed towards the inside of the polygon

- The reference $x_{sw,des}$ for the swing feet is computed using two splines: the former to lift the foot, the latter to lower it
- Considering $T_{sw} > 0$ the duration of the swing phase, each spline lasts $0{,}5T_{sw}$



Motion planner

- The motion planner can be implemented using the TOWR library (https://github.com/ethz-adrl/towr)
- The TOWR library computes the CoM reference and the swing feet reference based on:
  - A desired gait
  - The kinematic limits of the used robot
  - The desired trajectory

Momentum-based observer

- External forces are estimated to compensate for them inside the controller
  - They can reconstruct unknown forces that may arise from several reasons such as unmodelled or inaccurate model parameters, external pushing actions, collisions with obstacles, and so on
- In order to take into account the disturbances acting on both stance and swing legs, the observer considers the momentum of all the legs

$$\rho = M_{22}\dot{q}_j$$

- Considering the part of the motion equations related to the legs, the time derivative of ρ is

$$\dot{\rho} = \bar{C}_2^T\dot{q}_j + \tau + \bar{J}_{st,j}^T f_{gr} + \bar{J}_j^T f_{ext}$$

- Define $\hat{F} = \bar{J}_j^T\hat{f} \in \mathbb{R}^{n_j}$ and $F_{ext} = \bar{J}_j^T f_{ext} \in \mathbb{R}^{n_j}$
  - F_{ext} represents the effect at the joint torques of the resultant force at the legs' tips
  - $\hat{F}$ represents the estimated joint torques corresponding to the resultant force at the legs' tips
- The estimator is explicitly designed to achieve a linear relationship between the estimated external forces and the real ones in the Laplace domain

$$\hat{F} = G(s)F_{ext}$$

- The estimator is designed in the following way

$$\gamma_1(t) = K_1(\rho(t) - \int_0^t \left(\hat{F}(\sigma) + \alpha(\sigma)\right) d\sigma)$$

$$\gamma_i(t) = K_i \int_0^t \left(-\hat{F}(\sigma) + \gamma_{i-1}(\sigma)\right) d\sigma) \qquad i = 2, \dots, r$$

  - where $\hat{F} = \gamma_r$, $K_i \in \mathbb{R}^{(n_j)\times(n_j)}$ are positive definite gain matrices, $r > 0$ is the desired degree of the estimator and

$$\alpha(t) = \bar{C}_2^T \dot{q}_j + \tau + \bar{J}_{st,j}^T f_{gr}$$

- The desired acceleration and the desired ground reaction forces are obtained through a quadratic problem

$$\begin{aligned} \underset{\zeta}{\text{minimize}} \quad & f(\zeta) \\ \text{subject to} \quad & A\zeta = b, \\ & D\zeta \leq c. \end{aligned}$$

  - The constraints regard
    - Dynamic consistency
    - Non-sliding contact
    - Torque limits
    - Swing legs task
  - The chosen vector of control variables is

$$\zeta = [\ddot{r}_c^T \ \ \ddot{q}_j^T \ \ f_{gr}^T]^T \in \mathbb{R}^{n_b+n_j+3n_{st}}$$

    - where $r_c = [\,p_c^T \ \ \eta_c^T]^T$

- Cost function
  - It aims at tracking the CoM's reference
  - To this aim, it is useful to consider the first n_b equations of the centroidal dynamics.

$$M_{CoM}(q)\ddot{r}_c + mg = W_{CoM} = \bar{J}_{st,c}^T f_{gr} + \bar{J}_c^T f_{ext}$$

    - where $M_{CoM} = \begin{bmatrix} mI_3 & O \\ O & M_{11} \end{bmatrix}$.
    - $W_{CoM} \in \mathbb{R}^6$ is the wrench at the robot's CoM including inertial and gravity terms
  - The reference of the CoM is compliantly achieved defining the following desired wrench at the CoM

$$W_{CoM,des} = -K_p(r_c - r_{c,ref}) - K_d(\dot{r}_c - \dot{r}_{c,ref}) + mg + M_{CoM}(q)\ddot{r}_{c,ref}$$

    - with $K_p, K_d \in \mathbb{R}^{6\times 6}$ positive definite gain matrices
    - $-K_p(r_c - r_{c,ref}) - K_d(\dot{r}_c - \dot{r}_{c,ref})$ represents a Cartesian impedance
    - mg is the gravity compensation
    - $M_{CoM}(q)\ddot{r}_{c,ref}$ is a feedforward term
  - A suitable compliant behavior has to be ensured for this kind of robot to get bounded interaction forces
    - Implementing a Cartesian impedance to stabilize the CoM motion, a compliant balancing for the robot is achieved
  - K_p and K_d represent the stiffness and the damping terms of the impedance, respectively.
  - Adding the gravity compensation an impedance control with gravity compensation is obtained , like a PD+ controller
    - It allows precise tracking of the reference in free motion and results in a passive system for interaction with the environment

- The cost function can be written as

$$f(\zeta) = \left\| \bar{J}_{st,c}^T \Sigma \zeta + \bar{J}_{st,c}^T \hat{f}_{st} - W_{CoM,des} \right\| + \|\zeta\|$$

  - with $\Sigma \in \mathbb{R}^{n_b+n_j+3n_{st}}$ a matrix selecting the last $3n_{st}$ elements of ζ, the ground reaction forces
- The cost function is designed in order to track the desired wrench

- Dynamical consistency (equality constraints)
  - The centroidal dynamic is imposed in the absence, or perfectly compensated, of external disturbances
  $$M_{CoM}(q)\ddot{r}_c + mg = \bar{J}_{st,c}^T f_{gr}$$
    - Using the control variables ζ the constraint can be written as
    $$\begin{bmatrix} M_{CoM}(q) & O_{n_b \times n_j} & -\bar{J}_{st,c}^T \end{bmatrix} \zeta = -mg$$
  - A contact constraint at the acceleration level is imposed for all the stance feet
  $$\bar{J}_{st,c}\ddot{r}_c + \dot{\bar{J}}_{st,c}\dot{r}_c + \bar{J}_{st,j}\ddot{q}_j + \dot{\bar{J}}_{st,j}\dot{q}_j = 0_{3n_{st}}$$
    - Using the control variables ζ the constraint can be written as $\begin{bmatrix} \bar{J}_{st,c} & \bar{J}_{st,j} & O_{3n_{st} \times 3n_{st}} \end{bmatrix} \zeta = -\dot{\bar{J}}_{st,c}\dot{r}_c - \dot{\bar{J}}_{st,j}\dot{q}_j$

- Non-sliding contact (Inequality constraints)
  - Ground reaction forces need to be constrained inside a friction cone to avoid slipping
  - For control design purposes, the friction cone can be approximated as a pyramid to obtain linear constraints in the optimization problem
  - For the *i*-th foot, the friction pyramid constraint can be written as

$$(\bar{l}_{1,i} - \mu \bar{n}_i)^T f_{gr,i} \leq 0$$
$$-(\bar{l}_{1,i} + \mu \bar{n}_i)^T f_{gr,i} \leq 0$$
$$(\bar{l}_{2,i} - \mu \bar{n}_i)^T f_{gr,i} \leq 0$$
$$-(\bar{l}_{2,i} + \mu \bar{n}_i)^T f_{gr,i} \leq 0$$

    - where $\bar{n}_i$ is the *i*-th normal vector to the contact ground, while $\bar{l}_{1,i}$ and $\bar{l}_{2,i}$ are two tangential vectors lying on the same surface
  - Using the control variables ζ the constraint can be written as

$$\begin{bmatrix} O_{4n_{st} \times n_b} & O_{4n_{st} \times n_j} & D_{fr} \end{bmatrix} \zeta \leq 0_{4n_{st}}$$

    - where $D_{fr} \in \mathbb{R}^{4n_{st} \times 3n_{st}}$ is a diagonal matrix containing the friction cone constraints expressed for each stance leg

$$D_{fr} = \begin{bmatrix} D_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & D_{n_{st}} \end{bmatrix} \quad \text{with} \quad D_i = \begin{bmatrix} (\bar{l}_{1,i} - \mu \bar{n}_i)^T \\ -(\bar{l}_{1,i} + \mu \bar{n}_i)^T \\ (\bar{l}_{2,i} - \mu \bar{n}_i)^T \\ -(\bar{l}_{2,i} + \mu \bar{n}_i)^T \end{bmatrix}$$

Torque limits (Inequality constraints)

- For mechanical and safety reasons, joint torques need always to be limited
- Being $\tau_{min}, \tau_{max} \in \mathbb{R}^{n_j}$ the minimum and maximum reachable torques, respectively, considering the part of the motion equations regarding robot's legs, the constraints about limited torques can be expressed as follow

$$\tau_{min} - \bar{C}_2 \dot{q}_j \leq M_{22} \ddot{q}_j - \bar{J}_{st,j}^T f_{gr} \leq \tau_{max} - \bar{C}_2 \dot{q}_j$$

- Using the control variables ζ the constraint can be written as

$$\begin{cases} [\, O_{n_j \times n_b} \quad M_{22} \quad -\bar{J}_{st,j}^T]\zeta \leq \tau_{max} - \bar{C}_2 \dot{q}_j \\ [\, O_{n_j \times n_b} \quad -M_{22} \quad \bar{J}_{st,j}^T]\zeta \leq -(\tau_{min} - \bar{C}_2 \dot{q}_j) \end{cases}$$

- Swing leg task (Inequality constraints)
  - It allows the robot to follow the trajectory planned for the swing feet
  - Recall the dynamically consistent support null-space matrix as

$$N_c = I_{n_{st}} - J_{st}^{\#} J_{st}$$

  - Premultiply the equations of motion by N_c

$$N_c(\bar{M}\dot{v}_c + \bar{C}v_c + G) = N_c S^T \tau + N_c J_{sw}^T f_{sw,ext}$$

    - where $G = \begin{pmatrix} mg_0 \\ 0_3 \\ 0_{n_j} \end{pmatrix}$.
  - Remembering that N_c is an orthogonal projection operator, such that $N_c J_{st}^T = 0$, it is worth noticing that the only remaining term related to external forces is $J_{sw}^T f_{sw,ext}$, which regards the swing legs

- The resulting equation can be written as

$$M_c \dot{v}_c + N_c \bar{C} v_c + N_c G - P v_c = N_c S^T \tau + N_c J_{sw}^T f_{sw,ext}$$

  - where $M_c = N_c \overline{M} + I_{n_b+n_j} - N_c$ and $P = -J_{st}^{\#} \dot{J}_{st}$
  - M_c is always invertible, provided that M is invertible
- Let x_{sw} be the position of the swing feet: premultiplying the equation by $J_{sw} M_c^{-1}$, the following operational space configuration for the swing legs can be recovered

$$\ddot{x}_{sw} - \dot{J}_{sw} v_c + J_{sw} M_c^{-1} (N_c \bar{C} v_c + N_c G - P v_c) = J_{sw}\ M_c^{-1} N_c S^T \tau + J_{sw} M_c^{-1} N_c J_{sw}^T f_{sw,ext}$$

  - where the following relation has been employed

$$\ddot{x}_{sw} = J_{sw} \dot{v}_c + \dot{J}_{sw} v_c$$

- To compensate for disturbances, the term related to external forces on swing legs in the equation must be taken into account
- Considering the reference for the swing feet $x_{sw,des}$, $\dot{x}_{sw,des}$, and $\ddot{x}_{sw,des}$, the command acceleration for the swing feet can be written as

$$\ddot{x}_{sw,cmd} = \ddot{x}_{sw,des} + K_{d,sw}\left(\dot{x}_{sw,des} - \dot{x}_{sw}\right) + K_{p,sw}\left(x_{sw,des} - x_{sw}\right) - J_{sw} M_c^{-1} N_c J_{sw}^T \hat{f}_{sw}$$

- To follow the trajectory, the following equality constraint should be imposed

$$\bar{J}_{sw,c} \ddot{r}_c + \dot{\bar{J}}_{sw,c} \dot{r}_c + \bar{J}_{sw,j} \ddot{q}_j + \dot{\bar{J}}_{sw,j} \dot{q}_j = \ddot{x}_{sw,cmd}$$

- The constraint is softened by adding slack variables $\gamma \in \mathbb{R}^{3n_{sw}}$ within the optimization problem
- The addressed inequality constraint is thus chosen as

$$\ddot{x}_{sw,cmd} - \gamma \leq \bar{J}_{sw,c}\ddot{r}_c + \dot{\bar{J}}_{sw,c}\dot{r}_c + \bar{J}_{sw,j}\ddot{q}_j + \dot{\bar{J}}_{sw,j}\dot{q}_j \leq \ddot{x}_{sw,cmd} + \gamma$$

- For this reason, the control variables vector becomes $\zeta = [\ddot{r}_c^T \quad \ddot{q}_j^T \quad f_{gr}^T \; \gamma^T]^T$ and the constraint can be written as

$$\begin{cases} [\bar{J}_{sw,c} \quad \bar{J}_{sw,j} \quad 0_{3n_{sw}\times 3n_{st}} \quad I_{3n_{sw}\times 3n_{sw}}]\zeta \leq \ddot{x}_{sw,cmd} - \dot{\bar{J}}_{sw,c}\dot{r}_c - \dot{\bar{J}}_{sw,j}\dot{q}_j \\ [-\bar{J}_{sw,c} \quad -\bar{J}_{sw,j} \quad 0_{3n_{sw}\times 3n_{st}} \quad -I_{3n_{sw}\times 3n_{sw}}]\zeta \leq -\ddot{x}_{sw,cmd} + \dot{\bar{J}}_{sw,c}\dot{r}_c + \dot{\bar{J}}_{sw,j}\dot{q}_j \end{cases}$$

- Therefore, the matrices A, b, D and c of the quadratic problem can be written as

$$A = \begin{bmatrix} M_{CoM}(q) & O_{n_b \times n_j} & -\bar{J}_{st,c}^T & O_{n_b \times 3n_{sw}} \\ \bar{J}_{st,c} & \bar{J}_{st,j} & O_{3n_{st} \times 3n_{st}} & O_{3n_{st} \times 3n_{sw}} \end{bmatrix}$$

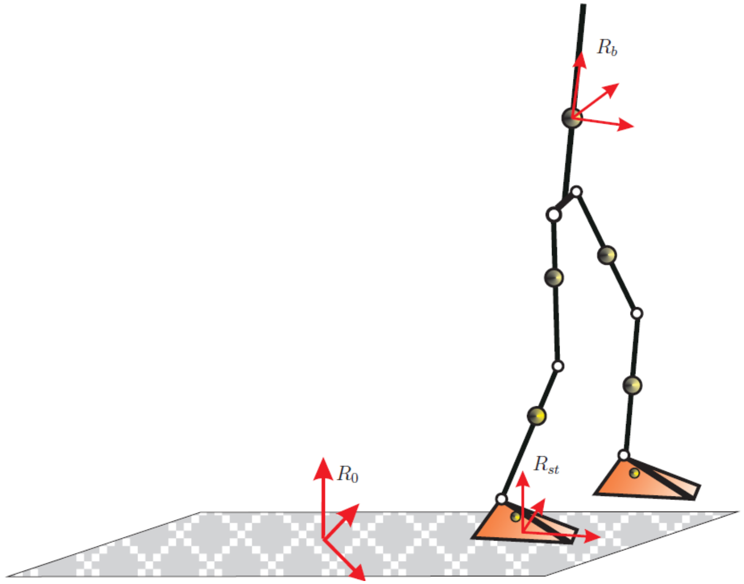
$$b = \begin{bmatrix} -mg \\ -\dot{\bar{J}}_{st,c}\dot{r}_c - \dot{\bar{J}}_{st,j}\dot{q}_j \end{bmatrix}$$

$$D = \begin{bmatrix} O_{4n_{st} \times 6} & O_{4n_{st} \times n_j} & D_{fr} & O_{4n_{st} \times 3n_{sw}} \\ O_{n_j \times n_b} & M_{22} & -\bar{J}_{st,j}^T & O_{n_j \times 3n_{sw}} \\ O_{n_j \times n_b} & -M_{22} & \bar{J}_{st,j}^T & O_{n_j \times 3n_{sw}} \\ \bar{J}_{sw,c} & \bar{J}_{sw,j} & 0_{3n_{sw} \times 3n_{st}} & I_{3n_{sw} \times 3n_{sw}} \\ -\bar{J}_{sw,c} & -\bar{J}_{sw,j} & 0_{3n_{sw} \times 3n_{st}} & I_{3n_{sw} \times 3n_{sw}} \end{bmatrix}$$

$$c = \begin{bmatrix} 0_{4n_{st}} \\ \tau_{max} - \bar{C}_2\dot{q}_j \\ -(\tau_{min} - \bar{C}_2\dot{q}_j) \\ \ddot{x}_{sw,cmd} - \dot{\bar{J}}_{sw,c}\dot{r}_c - \dot{\bar{J}}_{sw,j}\dot{q}_j \\ -\ddot{x}_{sw,cmd} + \dot{\bar{J}}_{sw,c}\dot{r}_c + \dot{\bar{J}}_{sw,j}\dot{q}_j \end{bmatrix}$$

- Biped robots are traditionally modeled as tree structures
- The collisions between the feet and the ground are also assumed to be rigid ones
- All the previous assumptions on contacts and friction hold
- The feet can be considered as point-like or, more frequently, flat-soled
- While walking, the robot can either have one foot on the ground or two feet on the ground: the first scenario is called *single support,* while the second one is called *double support.*
- Depending on the type of foot and the type of support, the dynamic model may be subject to some simplifications

- Assuming a single support stance, the foot must have at least three non-collinear points in contact with the ground, and the GRF can be divided in its wrench (GRW) and moment (GRM) components.
- The GRM is non-null only if the force is applied askewly from the foot origin. The point on the floor in which the GRM components around the x and y axes are null is the Zero Moment Point
- Given the single support and the stable walking hypotheses, it is possible to adapt the complementarity dynamic model to include the GRW, as for static walking it is assumed that the ZMP falls into the support polygon.
- Let $v_{st}^{\delta t} \in \mathbb{R}^3, \omega_{st}^{\delta t} \in \mathbb{R}^3$ be the linear and angular velocity of the stance feet, and let $T_{st}(q_c) \in \mathbb{R}^{4\times4}$ be the homogeonous transformation of the stance foot in the world frame.

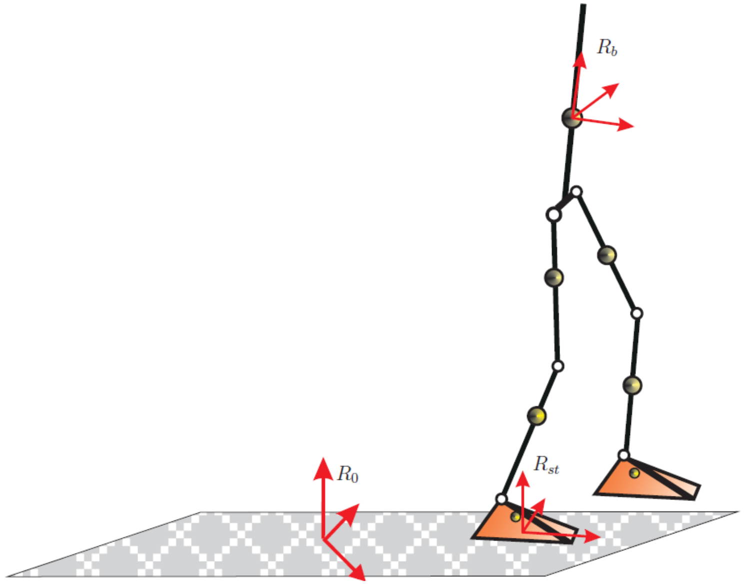


- The equation thus becomes

$$\bar{M}(q_c)\dot{v}_c + \bar{C}(q_c, v_c)v_c + \bar{g}(q_c) = S^T\tau + J_{st}^T(q_c)F_{st}$$

- It is also possible to write

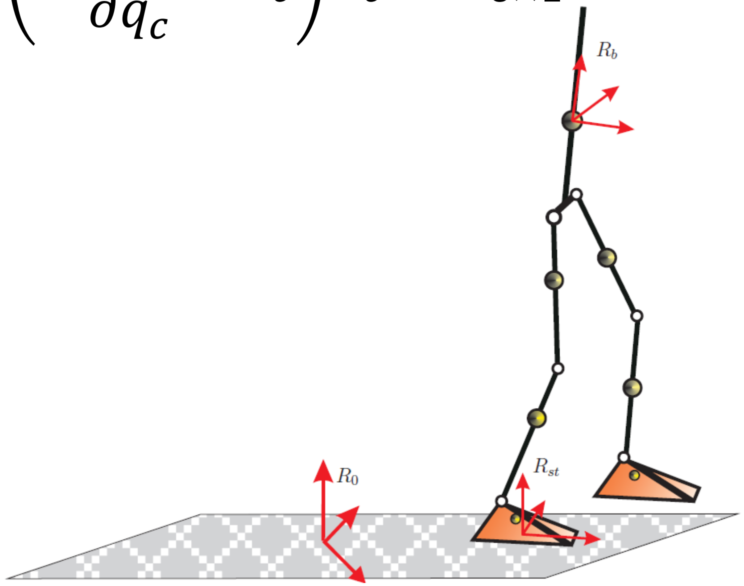
$$\begin{bmatrix} v_{st}^{\delta t} \\ \omega_{st}^{\delta t} \end{bmatrix} = J_{st}(q_c)v_c$$



- Assuming the foot to be fixed to the ground, and considering a reduced set of coordinates for the foot position and orientation $q_{st}(q_c)$, it is possible to show that its jacobian is full rank. This implies that the reduced coordinates can be written as a constant vector. From this we have

$$\begin{bmatrix} v_{st}^{\delta t} \\ \omega_{st}^{\delta t} \end{bmatrix} = J_{st}(q_c) v_c = 0_{6\times 1}, \qquad J_{st}(q_c)\dot{v}_c + \frac{\partial}{\partial q_c}\left(\frac{\partial q_{st}(q_c)}{\partial q_c} v_c\right) v_c = 0_{6\times 1}$$

- So both the velocities and the accelerations of the stance foot are null. Moreover, combining the second equation with the dynamic model allows for the extraction of both $\dot{v}_c$ and $F_{st}(q_c, v_c, \tau)$.



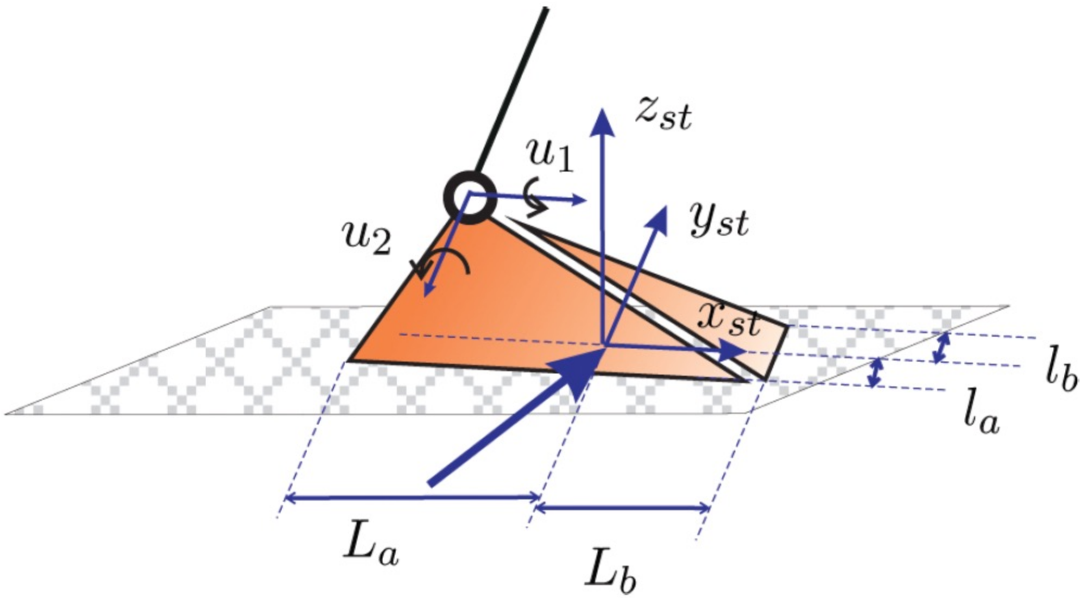
- Since we have assumed that the foot is fixed to the ground, all the assumptions that were made regarding the friction model must be satisfied. We then have

$$\begin{cases} f_z \geq 0 \\ |f_x| \leq \frac{\sqrt{2}}{2}\mu f_z \\ |f_y| \leq \frac{\sqrt{2}}{2}\mu f_z \end{cases}$$

- The foot may push onto the ground but at the same time rotate along one of its edges. For this reason, two other conditions must be met:

$$-l_b f_z < m_x < l_a f_z$$
$$-L_a f_z < m_y < L_b f_z$$

- where m_x, m_y are the moments around the relative axes



- The previous constraints can be grouped in the single constraint $A_{F_{st}}^T(q_c)F_{st}(q_c, v_c, \tau) > 0$.
- Adding a unilateral constraint to account for the distance of the impact $h(q_c) > 0$ allows for the expression of the complete constraint system

$$H(q_c, v_c, \tau) = \begin{bmatrix} A_{F_{st}}^T(q_c)F_{st}(q_c, v_c, \tau) \\ h(q_c) \end{bmatrix} > 0$$

- When these conditions hold, the robot model can be simplified as just a function of the joint variables

$$M(q_j)\,\ddot{q}_j + C(q_j, \dot{q}_j)\dot{q}_j + G(q_j) = S^T\tau$$

- Keep in mind that the two constraints on the rolling of the foot, which constrain the position of the Zero Moment Point, are the harder ones to satisfy and to verify, and that this difficulty increases as the foot size decreases.

- During a walking gait, it is generally true that the swing foot hits the ground with a non-zero velocity
- Assuming that the impact is rigid, the contact wrench acts in an instant and is thus modeled as an impulse. This impulse brings the foot to a halt, causing also a discontinuity in the velocity and an impulsive acceleration

$$\bar{M}(q_c)\dot{v}_c + \bar{C}(q_c, v_c)v_c + \bar{g}(q_c) = S^T\tau + J^T(q_c)\delta f_{imp}$$

- Integrating it over the duration of the impact gives us

$$\bar{M}(q_c)(v_c^+ - v_c^-) = J^T(q_c)F_{imp} = J^T(q_c)\int_{t^-}^{t^+}\delta f_{imp}(\tau)d\tau$$

- We assume v_c^+, v_c^- to be the velocity after and before the impact. Moreover, it is assumed that $q_c^+ = q_c^- = q_c$. The combination of these equations expresses the *conservation of generalized momentum.*

- Notice that the velocity before the impact is known, while the one after the impact isn't, as is the total force exerted at impact. This means that we need more equations to get all the data.
- Some key assumptions are made for this reason:
  - The impact of the swing leg occurs wih the sole on a flat surface
  - The foot neither slips nor rotates at impact
  - The stance foot is released immediately after the impact
- This last condition is translated into having an instantaneous double support phase, which can be expressed via a constraint on the swing leg $J(q_c)v_c = 0$. By combining this with the *conservation of generalized momentum*, we obtain

$$\begin{bmatrix} \bar{M}(q_c) & -J^T(q_c) \\ J(q_c) & 0 \end{bmatrix} \begin{bmatrix} v_c^+ \\ F_{imp} \end{bmatrix} = \begin{bmatrix} \bar{M}(q_c)v_c \\ 0 \end{bmatrix}$$

- In the previous equation, the first matrix is full rank as long as the J matrix is full rank
- From this it is now possible to extract the value of the post-impact velocity

$$v_c^+ = \left(I_{n+6} - \bar{M}^{-1} J^T \left(J \bar{M}^{-1}\right) J\right) v_c$$

- Then, projecting down the result from the extended model to the reduced joint one gives

$$\dot{q}_j{}^+ = \Delta(q_j) \dot{q}_j{}^-$$

- This expression of the velocities is standard notation in many textbooks.