Literature Review

Romeo Orsolino

February 8, 2016

Contents

Abstract

This is a collection of the literature review of my PhD. I wrote a small summary for each paper I read and I consider interesting for the topic of dynamic locomotion.

1 Kalakrishnan, Buchli, Mistry, Schaal - Learning, Planning and control for quadruped locomotion over challenging terrain

2 Tedrake, Wieber - Modeling and Control of Legged Robots [1]

Authors: Wieber, Tedrake, Kuindersma
Year: 2015

Summary

This documents develops a comprehensive summary regarding the main tools available to design dynamic gaits for walking robots (biped robots specifically). A particular focus is given to the controllability of the gait and the robustness analysis; the following four stability measures are explained:

- Robust stability
- Stochastic stability
- Input-Output stability
- Stability margin: keeping away from boundaries such as the distance between the CoP and the edge of the convex support polygon.

The control design has got the following main tools for the robustness analysis of a given gait:

- Fixed points: A fixed point is a safe configuration of the robot where it can stand still;
- Limit cycles: it represents an extension of the fixed point to the case of periodic gaits. More precisely it is a periodic orbit, namely a solution $\{q(t), u(t), f(t) | t \in [0, \infty)\}$ such that $q(t+T) = q(t)$, $u(t+T) = u(t)$ and $f(t+T) = f(t)$ where T is the period of the periodic orbit in the phase plane. A periodic solution is (asymptotically) orbitally stable if:

  $$\lim_{t \to \infty} [\min_{0 \leq t' \leq T} ||q(t) - q_0(t')||] = 0$$

  This means that the trajectories must not converge, only the distance between the initial point and the closest point must go to zero. This can be defined in a region containing $q_0(.)$ or globally (even if globally stable legged robots are of little interest). The existence of such limit cycle depends on the actuation of the system, for a highly underactuated (or even passive) robot the computation of a limit cycle can be very challenging. As an opposite, for fully actuated robots, there exist many control policies $\mathbf{u}(t) = \boldsymbol{\pi}(t, \mathbf{q}(t), \dot{\mathbf{q}}(t), \mathbf{f}(t)$ that will lead to a limit cycle.
- Viability = viability is referred to as "not falling down" capability. The viable space is thus defined as the set of states where the robot is able to avoid to fall down.
- Controllability = capability to bring the robot from an arbitrary initial state to a final one within a finite time. In linear systems this is a property of the system, regardless of the initial and final conditions and it can be analysed by means of the controllability matrix. For nonlinear systems instead controllability is strongly dependent the initial and final states. As a consequence we can distinguish different types of controllability depending on the initial and final states. For legged robots the controllability is considered as the ability to bring the system to a stable fixed point (called **capture point**). The set of initial states that can lead to a fixed point within a limited number of steps is called **capturable**.

Key points / Takeaways

1. definition of the Capture Point CP as a stable fixed point of the Poincar Map
2. Clear description of the coupled dynamics of CoM, ZMP and CP.

Weak points

Ideas

1. They should be better highlighted and fully described the links between CP and poincar maps
2. Possibly the concept of CP should be extended to the ability to pass from a limit cycle to another limit cycle.
3. Worst case analysis of controller robustness has been given little attention so far. An extension of the Lyapunov functions-based methods shows that "nominal solutions of the limit cycle are often parameter dependent".

3 Tedrake - Underactuated Robotics

4 Park - Online Planning for autonomous running jump over obstacles in high-speed quadrupeds

5 J. Pratt, Goswami - Capture Point: A Step toward Humanoid Push Recovery [2]

Authors: J. Pratt, Goswami
Year: 2006

Summary

The Capture Point CP is defined for push recovery purposes, namely for determining how many steps must be performed by the biped robot to recover from a push. This is done on a simplified LIP + Flywheel model as the whole body dynamics of a robot is too complicated (being it highly dimensional, nonlinear and **hybrid**). First of all the CP is computed using the LIP model only, so that the robot can only act on the linear force f_k for the push recovery, this leads to a unique point CP. The addition of the Flywheel acts as a further degree of freedom of the simplified model, that can use both f_k and τ to recover from the push. This leads to the computation of a **capture region**, namely the set of all the capture points. The different

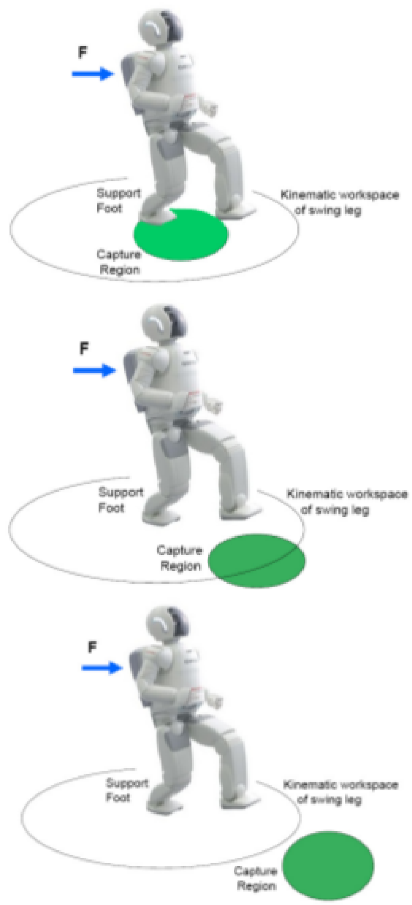

Figure 1: The biped can recover from a push when its support polygon intersects the capture region

capture regions are computed for the following cases:

- Pure LIP model
- LIP+Flywheel with instantaneous velocity variation
- LIP+Flywheel with instantaneous position variation

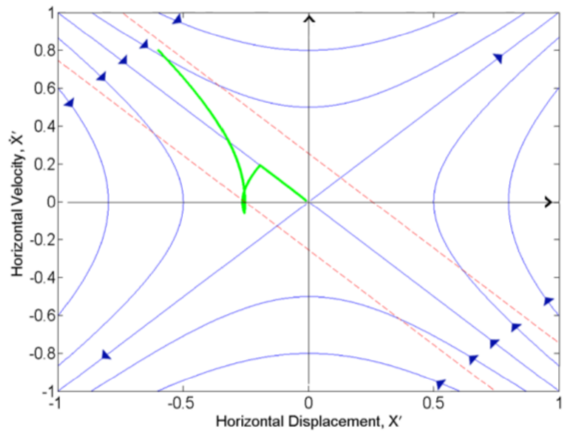

Figure 2: Phase portrait of the LIP + Flywheel model

- LIP+Flywheel with limited torque and angle

In figure 5 you can see here the trajectory of the x coordinate of the CoM starting from an arbitrary point and reaching a *capture state* performing a *safe feasible trajectory*. In the same figure also the capture region is represented with the red dotted line. You can see here that the green trajectory goes out of the capture region for a few instants, this is because the LIP+Flywheel as a non-null momentum in that points whereas the displayed capture region was computed for an initial state with null momentum.

Key points / Takeaways

1. Closed-form expression of the CP for a LIP + Flywheel model model
2. Determining which is the best lunge profile to recover from the push, solving for example an optimization problem
3. The analysis of the capture region is carried out also with non-dimensional variables in order to detect the minimum number of variables needed.

Weak points

- What do the blue curves in the phase plane of figure 5 represent? **Answer**: they represent the evolution of the system in the autonomous case, when no external input is applied to the robot.

Ideas

1. The definition of CP could be studied also for a SPRINGED LIP + Flywheel
2. How does the Phase Plane look like if we use the $(\dot{x}, \ddot{x})$ variables? can we maybe in this case define a limit cycle and use the Poincaré return map for checking the system stability?

6 J. Pratt, Goswami - Capturability Based Analysis and Control of Legged Robots - Part 1 [3]

Authors: J. Pratt, Goswami
Year: 2012

Summary

The Capture point theory is here explained for 3D-LIP model and for the 3D-LIPM model (with non zero moment of inertia) The *capturability* is updated at the end of every stance phase: there is no way to change the capturability during the swing phase.

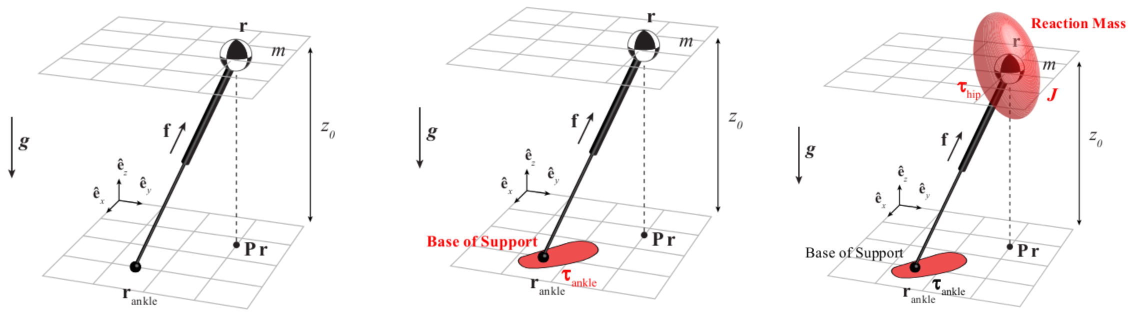

Figure 3: three simplified models of a the dynamics of an antropomorphic robot

3D-LIP model:

1. Equations of motion:

$$m\ddot{r} = mg + f$$

$$-(r - r_{ankle}) \times f = 0$$

2. CP definition:

$$r_{ic} = Pr + \frac{\dot{r}}{\omega}$$

3. Instantaneous CP dynamics:

$$\ddot{r} = \omega^2(Pr - r_{ankle})$$

4. 0-step capturability:

$$|r'_{ic} - r'_{ankle}| \leq d'_0 = 0$$

5. n-steps capturability:

$$|r'_{ic} - r'_{ankle}| \leq d'_n$$

where:

$$d'_n = (l'_{max} + d'_{n-1})e^{-\Delta t'_s}$$

3D-LIP + finite foot:

1. Equations of motion:

$$m\ddot{r} = mg + f$$

$$-(r - r_{ankle}) \times f + \tau_{ankle} = 0$$

2. CP definition:
no c.f. sol.

3. Instantaneous CP dynamics:

$$\ddot{r} = \omega^2(Pr - r_{CoP})$$

4. 0-step capturability:

$$|r'_{ic} - r'_{ankle}| \leq d'_0 = r'_{max}$$

5. n-steps capturability:

$$|r'_{ic} - r'_{ankle}| \leq d'_n$$

where: $d'_n = r'_{max} + (l'_{max} - r'_{max} + d'_{n-1})e^{-\Delta t'_s}$

3D-LIP + finite foot + reaction mass:

1. Equations of motion:

$$m\ddot{r} = mg + f$$

$$J\dot{\omega} = \tau_{hip} - \omega \times (J\omega)$$

$$-(r - r_{ankle}) \times f + \tau_{ankle} - \tau_{hip} = 0$$

2. CP definition: no c.f. sol.

3. Instantaneous CP dynamics:

$$\ddot{r} = \omega^2(Pr - r_{CMP})$$

$$\dot{\omega} = J^{-1}P\tau_{hip}$$

4. 0-step capturability:

$$|r'_{ic} - r'_{ankle}| \leq d'_0 = r'_{max} + |\Delta r'_{CMP}|_{max}$$

5. n-steps capturability:

$$|r'_{ic} - r'_{ankle}| \leq d'_n$$

where: $d'_n = r'_{max} + |\Delta r'_{CMP}|_{max} + (l'_{max} - r'_{max} + d'_{n-1})e^{-\Delta t'_s}$

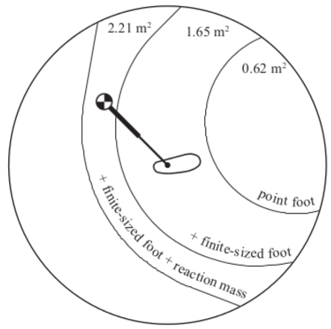

Figure 4: Comparison of the infinity-capturability distance d_∞ for the three different considered models

where:

- P is the matrix which projects the CoM on the ground.
- r_{max} = distance between the CoP and the point on the edge of the support which is closest to the istantaneous CP.
- CMP = Centroidal Moment Pivot point
- d_∞ = distance to the ∞-steps capturability region. This can be used as a **capturability margin**, in the sense that if the d_∞ takes on a small value, the robot will likely fall when subject to an external disturbance. For example, using the same common parameters it was obtained that for:
  - 3D-LIP model: $d_\infty = 0.431$
  - 3D-LIP model + finite foot: $d_\infty = 0.631$
  - 3D-LIP model + finite foot + reaction mass: $d_\infty = 0.664$

  So we can conclude that the third model has got the *highest level of capturability*, i.e. it is the most stable (see figure 4).

Key points / Takeaways

- **Equivalent constant CoP** =

Weak points

Ideas

7 J. Pratt, Goswami - Capturability Based Analysis and Control of Legged Robots - Part 2 [3]

Authors: J. Pratt, Goswami
Year: 2012

Summary

The *capturability* is updated at the end of every stance phase: there is no way to change the capturability during the swing phase.

Key points / Takeaways

Weak points

Ideas

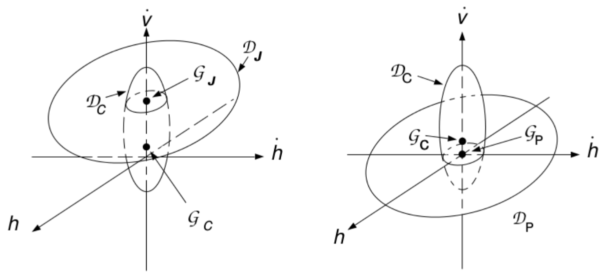

Figure 5: Representation of the three different domains of the three controllers: juggle, catcher and palmer. The palmer and juggler are two "modes" which belong to the same family, as described by Kris Hauser in [12]

8 D.E. Koditscheck - Sequential Composition of Dynamically Dexterous Robot Behaviors

Authors: D.E. Koditschek
Year: 2015

Summary

- Preimage backchaining concept is defined as inspired by [Lozano-Perez,Mason,Taylor 1984].
- In this paper a clear approach to the problem of hybrid systems is given. An hybrid control problem is here defined as the problem of defining a control for a mix of continuous and discrete systems.
- The coupled robot-environment state must be considered, not just one.
- "Event driven" trajectory generators are needed.
- The concept of invariant ellipsoid is given as the ellipsoid in the phase plane where the ball can be safely sent and there it does not lead to failure
- Basically three different controllers are defined: juggling, catching and palming. Each of them is defined on a different sub-domain and a passage from one to another is made possible just thanks to a higher-level controller which sets the path to the goal.

Key points / Takeaways

1. A *pre-image*, in the paper of Lozano-Perez, can be seen as the set of all the points of the space that allow to reach a goal with a single discrete motion
2. The term *back chaining* refers to process of splitting the trajectory from the goal back to the starting point in many *pre-images* that corrispond to discrete movements.
3. It is shown that the vertical velocity $\dot{z}$ has little influence when compared to the general horizontal speed $\dot{h}$ (either $\dot{x}$ or $\dot{y}$).

Weak points

Ideas

1. Extend the idea of *self-manipulation*: what if we reverse the problem of locomotion and we look at it as a manipulation problem? It is like considering that the earth is moving and the robot is a fixed frame. The manipulation consists in this case of many impacts which keep the earth in the "controllable region".

2. analyse a way to approach the control of **Discrete Events Systems**. In between two following events the system can thus be seen as an **autonomous system** (we cannot change the trajectory of the CoM between two events, what we can do is only to adjust the angular orientation and the position of the feet at the next impact).
3. From the manipulation point of view: we deal with the control of a DES when we have an obstacle which disconnects the workspace.

9 D.E. Koditschek - The Penn Jerboa: a platform for exploring parallel composition of templates - Koditschek

Authors: D.E. Koditschek
Year: 2015

Summary

Key points / Takeaways

Weak points

Ideas

10 D.E. Koditschek - Legged Self-manipulation [4]

Authors: D.E. Koditschek and Aaron M. Johnson
Year: 2013

Summary

Key points / Takeaways

Self-manipulation is a distinguished branch of locomotion which aims at exploiting the well mature field of manipulation. For this reason self-manipulation differs from both traditional manipulation and from locomotion for a few reasons; the main difference from traditional manipulation are the following:

- in self-manipulation the robot is the object and in the same time it is also the palm; therefore these two reference frames (palm and object) are in this case coincident. This means that it's the robot manipulating itself in order to perform a given movement.
- we're mainly concerned with the motion of the robot rather than the object.
- in self-locomotion the dynamics of legs and trunk are not decoupled.

Self-manipulation neither coincides with locomotion though, in that locomotion usually aims at using *templates*the simplified dynamic models. Self-manipulation instead aims at finding the best *anchors* which exploit the actuators performances.
In this sense self-manipulation can be used to *anchor* and locomotion problem solved in the traditional lumped-mass based method.

Weak points

Ideas

11 Rao - Numerical Methods for Optimal Control [5] [6]

Authors: Rao
Year: 2010

Summary

We can distinguish between *trajectory optimization* and *optimal control* problems mainly involved in control problems. The first case the optimization variables are static parameters while in the second case they are functions. In both cases the goal is to optimize (minimize or maximise) an objective index (which is a function in the case of trajectory optimization and it is a functional in the case of optimal control).
In the category of optimal control we can distinguish the following typical problems to be solved:

1. Systems of nonlinear equations
2. Differential equations and integration of function
   - Time-marching
     - Multiple-step methods (implicit (such as Euler forward, Crank-Nicolson or explicit liek Euler backward))
     - Multiple-stage methods (**Runge-Kutta** "discretization " method)
       In this method we divide the period of interest subintervals of duration h (the smaller is h the better will be the approximation). In this way the period is "discretized". The typical Runge-Kutta method is a fourth order method, meaning that the value of each stage is computed by means of 4 parameters. This method is very good *locally* in that the approximation performance decrease faw away from the initial stage.
   - Collocation
     - Gauss
     - Radau
     - Lobatto

     This methods aim at approximating a function with a polynomial, the order of which is determined by the number of conditions we can impose. Each condition refers to a time instant. All the *collocation conditions* make up together a system where they are rewritten in the form of *defect conditions*. The solution of this system leads to a *simultaneous* computation of all the parameters (as an opposite to the *predictor-corrector* approach found in time-marching).
     - Orthogonal collocation: in this method the collocation points are roots of special family of polynomials named *orthogonal polynomials* (e.g Legendre polynomials and Chebyshev polynomials). When the points are chose in an orthogonal mater (such as in this method) then the approximation to a definite integral is much more precise than the normal collocation methods.
3. Nonlinear optimization / Nonlinear Programming (NLP)
   - Gradient-based methods
     - Sequential Quadratic Programming (SQP)
     - Interior-point (IP)
     - Barrier methods
   - Euristic methods (stochastics)
     - Genetic algorithms: each genetic algorithm is made of the following components: encoding, fitness, selection, crossover and mutation
     - Simulated annealing

  - Particle swarm optimization (PSO)

The first two points are instances of problems to be solved with **indirect methods** while the second and the third are to be solved with **direct methods**. The second point can be solved with both methods.

- indirect methods/transcription:
- direct methods/transcription (discretize then optimize):
  - control parametrization method
  - state and control parametrization method (such as direct collocation)

Key points / Takeaways

Weak points

Ideas

12 Wang - Model-Based Position and Force Controller for a Hydraulic Legged Robot [7]

Authors: Guangrong Chen , Junzheng Wang, Shoukun Wang, Jiangbo Zhao, and Wei Shen
year: 2016

Summary

The compliance controller is a mix of active and passive compliance. The **realized compliance** Z_r is defined as the ratio between the applied force/torque $F(s)$ and the resulting position error $e(s) = x_0(s) - x(s)$. It is proved that the quantity $Z_r(s)$ can also be written as:

$$Z_r(s) = Z_d(s) \cdot G_p^{-1}(s)$$

where $Z_d(s)$ is the desired compliance and $G_p(s)$ is the inner closed-loop transfer function.
The compression of the passive spring $p_e(s)$ is just a bit of the total compression $p(s)$:

$$p_e(s) = [1 + Z_r^{-1}(s) \cdot Z_e(s)]^{-1} \cdot p(s)$$

As regards the stability we can say that the passive part of the system (the passive spring) is stable if the all system is stable and viceversa. In order for this property to hold in this system we need the term $[1+Z_r^{-1}(s) \cdot Z_e(s)]^{-1}$ to be stable. This is verified if $K_r \geq min(K_e, K_p)$ where K_r is the realized impedance of the system, K_e is the one of the passive spring (the environment) and K_p is the one of the active part.

Key points/ Takeaways

- two main conditions for a compliance controller to be stable: the inner c.l. transfer function must be stable and $K_r \geq min(K_e, K_p)$ must hold true.
- in the system described in this paper, a quadruped robot with passive springs in its legs we can define the total compliance as mix of active and passive one. The passive compliance term is a pure stiffness K_e, while the active term is composed of stiffness K_r and damping B_r (also named *dissipation component*). The damping part is due to the friction of the hydraulic actuators.
- The Baud-rate on a CANBUS device refers to the rate (speed) at which data is transmitted on the network. This is typically expressed in kilobits-per-second (kbps). In this case the CAN BUS baud-rate is 1000 kbps, cioé 1 Mbps (The baud-rate of a EtherCat can be up to 100 Mbps)
- A method is proposed at page 4 to define the desired reference stiffness K_d given the vicious friction (viscous friction) B_r of the hydraulic actuators.
- The compliance control sensibly reduces the peak in the impact force and the settling time is also considerably reduced.
- The compliant behavior cannot be achieved by only active compliance control if the controller is not fast enough. The adjoint of a passive spring reduces the impact force peak and makes the impact duration longer, giving more time to the controller in order to react.
- The Swing Leg Retraction (SLR), even if so far it was not assessed how to generate it, it does clearly reduce the impact forces
- In the case of ideal stiff actuator we assume that the actuator can reach in one single sample the desired value. In this case the inner closed-loop transfer function and $G_p = I$ and the position error transfer function $E_p = 0$.

Weak Points

- Shouldn't the title mention the world "compliance"? e.g. "mix of active and passive compliance"
- Figure n. 2 is not explained at all
- At line 52 page 2 right column two words are missing so the meaning of the whole sentence is not clear
- It looks like the robot is equipped with a spring on the leg, so the final compliance is a mix of active and passive one. This should be stated more clearly in the intro
- Figure 6: The letter P does not appear so it is not straighforward to read the image. Also F is not represented in the image whilst H is drawn twice so it might have two different meanings. Also the *landing angle* α should be represented in this figure.
- is it "vicious friction" or "viscous friction" (referred to the damping factor of the hydraulic actuators)?
- How could they get the blue dotted curve in figure 11? Didi they have to lock the actuators to obtain it?
- it should better be defined what a **passive system** is. A passive system in general is defined as a system where there is energy dissipation (such as the damping of spring-damper or such as the induction). In the considered system the dissipation is given by the term B_r which represents the damping of the active spring.
- the compliant behavior can be usually achieved in two ways:
  1. the force info is multiplied by the inverse of the desired impedance in order to compute the corresponding compression of the joints (this is called **admittance** control);
  2. the compression $\Delta\theta$ and the joint velocity $\dot{\theta}$ are multiplied by a stiffness factor P_{gain} and a damping factor D_{gain} (this is referred to as **impedance** control)

  In both way the force info (or the compression and speed info) can either come from the end-effector only, or from each joint of the leg.

Ideas

13 Wang - A strategy for push recovery in quadruped robot based on reinforcement learning - 2015

14 Focchi, Semini et al. - Dynamic Torque Control of a Hydraulic Quadruped Robot [8]

Authors: Thiago Boaventura, Claudio Semini, Jonas Buchli, Marco Frigerio, Michele Focchi, Darwin G. Caldwell
year: 2012

Summary

Main features required to a quadruped robot controller:

- precise: able to perform accurate footholds placement
- fast: able to produce fast movements
- robust: needed against uncertainties and disturbancies
- compliant: needed to withstand impacts and crashes

The **whole-body control** is a model-based controller which operates through an inversion of the dynamics, this can be seen as a **force/torque control**. The **position control** will still be present as a lower level control but will have low gains and thus have a little influence on the lower higher level.
When we talk about stability in the field of dynamic motions we had better refer to **limit-cicle stability**.

Key points / Takeaways

- the controller can be described as a three-layer controller:
  - The outer layer is a position control in the joint space
  - the middle layer is a force/torque controller
  - the inner layer is a position/speed control in the operational space and a pressure controller

  All these quantities are linearized thanks to a feedback linearization which leads to a linearized input v, this is computed as a PID controller. This PID controller is also responsible for the compliance of the system: in order to keep safe the mechanical parts and the electronics from impacts the torque peaks are always kept at safe level underneath 160 Nm.

- The robustness of the controller is not only achieved by means of robust controllers but also with robust hardware. The flow of the valves must have a large bandwith so that it can meet the requirements of different gaits.

- An equation for force control for hydraulically actuated robots can be computed using the *Bernoulli* equation (force equilibrium) [15].

Weak Points

Ideas

15 Focchi, Semini et al. - Gain scheduling control for the hydraulic actuation of the HyQ robot leg

Authors: B. Cunha, Thiago and Claudio, Semini and Emanuele, Guglielmino and Michele, Focchi and Darwin G., Caldwell
Year: 2010

Summary

Key points / Takeaways

Weak points

Ideas

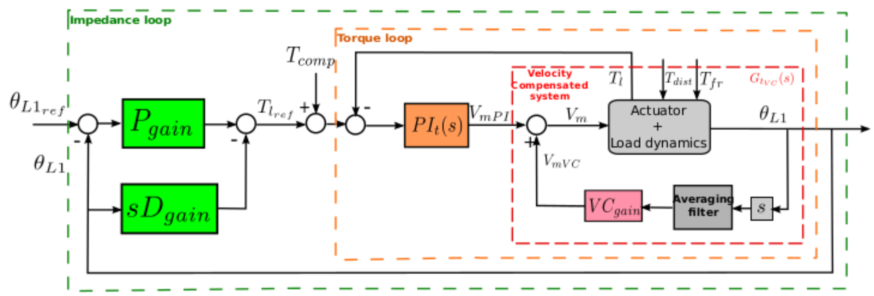

Figure 6: Block diagram of the velocity compensated system with inner torque loop and outer impedance loop

16 Focchi, Semini et al. - Robot Impedance Control and Passivity Analysis with Inner Torque and Velocity Feedback Loops [9]

Authors: Michele Focchi
Year: 2014

Summary

The interest of this paper is to identify the values of the parameters which yield a stable impedance in the system. In particular the focus is given to the HAA joint of HyQ.
The controller is a nested structure of three layers:

- inner positive feedback control loop (velocity compensation gain α). Here the velocity is obtained through derivative of the angular join position which yields highly noisy signals, as a consequence also a filter needs to be introduced, parametrized by N_{av}, to smooth the signal.
- torque loop (proportional feedback gain β)
- outer impedance control (PD controller with P_{gain} and D_{gain})

Besides the 5 parameters mentioned above also the sampling time t_s has got a high influence on the stability of the system.
The stability analysis is here carried out highlighting the **passivity** property of the system. Passivity is a conservative condition which states that given a passive system (such as an obstacle in the environment) the robot interaction with that system will be stable if it also behaves in a passive way. Passive springs are always passive in that they induce energy dissipation, but an active compliant system may be passive only under a proper analysis. In this paper it is shown that a **driving port impedance transfer function** $Z(s)$ will be positive only under the two following conditions:

1. $Z(s)$ has no pole with positive real part
2. the phase of $Z(s)$ lies between -90 and 90°

In the paper is also explained that, due to the derivative operator, the velocity compensator does not only influence the poles of the c.l.system (as the torque loop does), but it also affects the zeros of the system.

Key points / Takeaways

- compliance might be limited by the bandwidth of inner control loops (such as torque control loops by instance). The **bandwidth** is defined as the frequency where the torque amplitude decreases by -3dB with respect to the reference system.

- When we have a cascade of control loops it becomes fundamental to test the correct bandwidth of each loop. The bandwidth of the loop is connected to the time response, or more precisely, to the rise time in this way:

$$f(Hz) = \frac{1}{t_r(sec)}$$

  In general in cascaded control loops we have that the **bandwidth** of the innermost loop as to be at least one order of magnitude faster than the outer loop. The higher the bandwidth (faster system) the better the tracking error, whilst the lower the bandwidth (for slow systems) the higher the **damping effect**. In robotic systems where we are interested in achieving compliant behaviors, the bandwidth must be tuned in order to find a trade-off between **tracking** error and impedance.
  The bandwidth is directly affected by **sampling frequency** T_s and **filtering** N_{av} (number of samples to be filtered together): the lower the frequency of the system, the slower the system, the lower the bandwidth.
  Usually in **nested control loops** the slowest frequency acts as a **bottleneck** for the whole system.

- It is better to discretize the system and carry out the analysis in discrete domain because discretizing introduces many problems; for example a minimum phase system may become **non-minimum phase** when it is discretized.

- Sampling time of the controller: $t_s = 1$ ms

- The compliance controller can be seen as nothing but the tracking of the joint angle with more or less high PD gains. High P_{gain} and D_{gain} will result in stiff behavior (low compliance) while low gains will result in high compliance.

Weak points

Ideas

17 M. Diehl, Wieber - Fast direct multiple shooting algorithms for optimal robot control [10]

year: 2009

Optimal control problems can be devided in the following way:

- Dynamic Programming (Hamilton-Jacobi-Bellman Equation)
- Indirect Methods (Boundary Value Problem BVP, Pontryagin Maximum Principle)
- Dyrect methods (transform into a Nonlinear Programming Problem NLP)
    1. Single shooting
    2. Collocation
    3. Multiple shooting

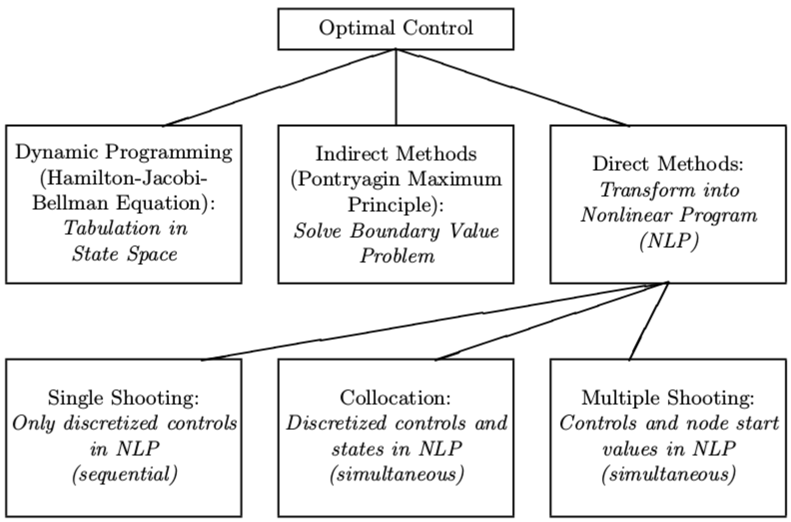

Figure 7: Different types of optimal control

Nonlinear Model Predictive Control (NMPC) can be defined as a feedback technique aimed at solving a system . The dynamics of the system must be thus expressed as an optimization problem. The optimal solution is repeatedly computed at every time step of length $\delta = n\epsilon$ where ϵ is the calculation time (CPU time) of a single SQL iteration and n is the number of need iterations to reach the convergence criterion.
Online Dilemma: A single SQP iteration (supposed it takes a constant time) can be solved in a time ϵ and we need n iterations before the optimal convergence criterion can be met. This means that the solution for the initial condition $\hat{x}(t_k)$ can be found only at the instant $t_k + n\epsilon$. The system state at that instant will be probably different from $\hat{x}(t_k)$ (in the best case the state won't have changed much) so it could be useful to try to predict the most likely state at that instant. Also disturbance rejection, for the same reason, is applied with a delay $\delta_d = \delta$
Real-time iteration scheme

18 Betts - Practical methods for optimal control and estimation using Nonlinear programming [11]

This stuff is very cool!

19 K. Hauser - Motion Planning for legged and humanoid robots [12]

year: 2008

Summary

The author defines the existence of different *modes* σ which belong to the same family Σ. The dimension of Σ, $dim(\Sigma)$, is lower than the dimension of the configuration space Q. All the modes of the same family are non-intersecting each other, meaning that if the robot is moving on a given trajectory q which is on the mode σ_1 and wants to move to the mode σ_2 (of the same family Σ), it will have to go through another family F which interesects the family Σ.

Key points / Takeaways

- PRM = Probabilistic Road Map
- MMPRM = Multi Modal Probabilistic Road Map
- family = foliation
- mode = leaves
- In legged locomotion a **mode** $\sigma \in \Sigma$ is a fixed set of footfalls. So σ stands for *stance* and Q_σ is the configuration space that can be reached by all the robots joints when the given stance σ is fixed. In general a mode σ defines a *feasible space* F_σ which is the set of configurations that satisfy the mode-specific constraints. Namely F_σ is a subset of Q (the configuration space of q), limited by the *dimensionality reducing constraints* and by the *volume reducing constraints*

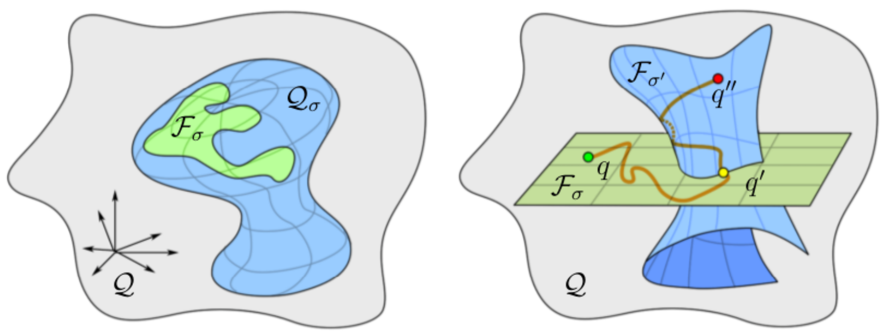

Figure 8: Family modes

Weak points

- The author distinguishes between *volume reducing constraints* and *dimensionality reducing contraints*. And how can we classify the nonholonomic contraints then?

ideas

20 K. Hauser, O. Ramon - Generalization of the Capture Point to Nonlinear Center of Mass and Uneven Terrain [13]

Year: 2015

Summary

This paper generalizes the concept of Capture Point as we know it based on a LIP model on a flat terrain. The new model is a Nonlinear Inverted Pendulum, meaning that the height of the CoM is no more constrained to be constant but it can be a steep line or a parabola. In the mean time also uneven terrains are considered, when characterized by a polygonal shape.

Key points / Takeaways

Weak points

Ideas

21 Boyd - Introduction to Linear Dynamical Systems

22 A. Bemporad - Model Predictive Control Course

23 A. Hof - The eXtrapolated Center of Mass concepts suggests a simple control of balance in walking [14]

Summary

The phase diagram given in the paper is useful in order to show that the equilibrium point (when the CoM is exactly above the CoP) is an unstable equilibrium in the sense that a small disturbance will make the state change to another configuration.
The Capture Point (or XCoM) can be seen as the

Key points/ takeaways

1. property 1: A sufficient condition for stability of the CoM trajectory is that the Xcom trajectory is stable (read: does not diverge too much compared to step length)
2. property 2: At any point the tangent to the trajectory of the Xcom runs through the CoP.

Weak points

24 P. Abbeel - Apprenticeship Learning and Reinforcement learning with application to robotic control - 2008

References

[1] R. Tedrake, P.B. Wieber, and S. Kuidersma. Modeling and control of legged robots. *Springer Handbook of Robotics*, chapter 48, 2015.

[2] Jerry Pratt, John Carff, Sergey Drakunov, and Ambarish Goswami. Capture point: A step toward humanoid push recovery. In *Proceedings of the Sixth IEEE-RAS International Conference on Humanoid Robots (Humanoids 2006)*, pages 200–207. IEEE, 2006.

[3] Twan Koolen, Tomas De Boer, John Rebula, Ambarish Goswami, and Jerry Pratt. Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models. *Int. J. Rob. Res.*, 31(9):1094–1113, August 2012.

[4] Aaron M Johnson and D E Koditschek. Legged self-manipulation. *IEEE Access*, 1:310–334, May 2013.

[5] Rao Anil V. A survey of numerical methods for optimal control. In *Advances in the astronautical sciences*, volume 315, 2010.

[6] Betts John T. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, chapter 4, pages 123–192.

[7] GuangRong Chen, Junzheng Wang, Shokun Wang, Jiangbo Zhao, and Wei Shen. Model-based position force controller for a hydraulic legged robot. 2016.

[8] Jonas Buchli Marco Frigerio Michele Focchi Darwin G. Caldwell Thiago Boaventura, Claudio Semini. Dynamic torque control of a hydraulic quadruped robot. *International Conference on Robotics and Automation*, 2012.

[9] Jonas Buchli Marco Frigerio Michele Focchi Darwin G. Caldwell Jonas Buchli Thiago Boaventura, Claudio Semini. Robot impedance control and passivity analysis with inner torque and velocity feedback. 2014.

[10] Moritz Diehl, Hans Georg Bock, Holger Diedam, and Pierre-Brice Wieber. Fast Direct Multiple Shooting Algorithms for Optimal Robot Control. In *Fast Motions in Biomechanics and Robotics*, Heidelberg, Germany, 2005.

[11] J.T. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming: Second Edition*. Advances in Design and Control. Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2010.

[12] K. Hauser. *Motion Planning for Legged and Humanoid Robots*. PhD thesis, 2008.

[13] O. Ramos and K. Hauser. *Generalization of the Capture Point for Nonlinear Center of Mass and Uneven Terrain*. 2015.

[14] Hof A. The extrapolated center of mass concept suggests a simple control of balance in walking. *Human Movement Science*, 27:112–125, November 2008.

[15] Cunha Thiago B., Semini Claudio, Guglielmino Emanuele, De Negri Victor J., Yang Yousheng, and Caldwell Darwin G. Gain scheduling control for the hydraulic actuation of the hyq robot leg. *International Conference on Robotics and Automation*, 2012.