Dynamic Trot-Walking with the Hydraulic Quadruped Robot - HyQ: Analytical Trajectory Generation and Active Compliance Control

Barkan Ugurlu, Ioannis Havoutis, Claudio Semini, and Darwin G. Caldwell

***Abstract*— This paper presents a trajectory generator and an active compliance control scheme, unified in a framework to synthesize dynamic, feasible and compliant trot-walking locomotion cycles for a stiff-by-nature hydraulically actuated quadruped robot. At the outset, a CoP-based trajectory generator that is constructed using an analytical solution is implemented to obtain feasible and dynamically balanced motion references in a systematic manner. Initial conditions are uniquely determined for symmetrical motion patterns, enforcing that trajectories are seamlessly connected both in position, velocity and acceleration levels, regardless of the given support phase. The active compliance controller, used simultaneously, is responsible for sufficient joint position/force regulation. An admittance block is utilized to compute joint displacements that correspond to joint force errors. In addition to position feedback, these joint displacements are inserted to the position control loop as a secondary feedback term. In doing so, active compliance control is achieved, while the position/force trade-off is modulated via the virtual admittance parameters. Various trot-walking experiments are conducted with the proposed framework using HyQ, a $\sim$ *75kg* hydraulically actuated quadruped robot. We present results of repetitive, continuous, and dynamically equilibrated trot-walking locomotion cycles, both on level surface and uneven surface walking experiments.**

I. Introduction

Decentralized dynamic legged locomotion control is often decomposed into two sub-phases: i) Online referential trajectory generation to obtain feasible motion inputs, ii) real-time sensory feedback control to regulate unmodelled dynamics and handle unexpected irregularities [1]. The robotics literature includes a sizable amount of research work, where this approach is experimentally proved to be efficient [2]–[9].

When it comes to quadrupedal trajectory generation, CPG (Central Pattern Generator) networks appear to be a popular choice. Researchers utilize nonlinear oscillators to synthesize trajectories with various shapes; such as, sinusoidal or elliptical [10]–[12]. CPGs are also observed in biological creatures; and therefore, their application to multi-legged locomotion is somewhat naturally insightful.

Despite their impressive performance, CPG-based motion generation has limitations. For example, often CPGs require non-intuitive hand-tuning while it is also difficult to incorporate on-line feedback, sometimes crucial for the success of the controller. Keeping these in mind, a group of researchers directed their attention to alternative approaches.

B. Ugurlu is with the Department of Brain Robot Interface, Computational Neuroscience Laboratories, ATR, 619-0288 Kyoto, Japan. I. Havoutis, C. Semini, and D. G. Caldwell are with the Department of Advanced Robotics, Istituto Italiano di Tecnologia, 16163 Genova, Italy. e-mail: barkanu@ieee.org; {ioannis.havoutis, claudio.semini, darwin.caldwell}@iit.it

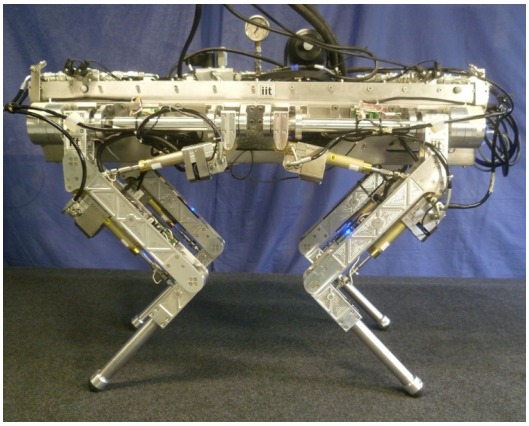

Fig. 1. The hydraulically actuated quadruped robot - HyQ. It has 12 degrees of freedom and its size is comparable to a goat. HyQ is designed for highly dynamic behavior, e.g. trotting and jumping.

For instance, Moro and his associates uses motion capture data from a horse and utilized these as kinematic motion primitives [13]. Pratt et al. make use of the capture point method for legged locomotion control [14]. Kalakrishnan and his teammates proposed a *Center of Pressure* (CoP)-based trajectory generation to achieve rough terrain locomotion [5].

Among these approaches, ZMP-based methods are proven to be reliable in generating physically viable quadrupedal locomotion trajectories [5], [9], [15]. The basic idea is to solve ZMP differential equations for a given feasible ZMP input to obtain the corresponding CoM trajectory (so-called inverse ZMP problem [9]). As noted by Byl et al. in [9], this approach cannot be implemented in a straightforward fashion, due to the nature of the mathematical equations that govern the ZMP model. To overcome this issue, Kajita implemented preview control, which can be interpreted as ZMP servo tracking control [8]. Byl et al. and Erbatur et al. particularly applied this approach to quadrupedal locomotion as well [9], [15].

While impressively useful in its own right, we believe that the ZMP-based feasible CoM motion generation can be solved by considering symmetry conditions, rather than interpreting the issue as a ZMP servo tracking problem. In doing so, this task can be realized in a simpler and more viable way without any necessity of ZMP feedback loop. One can generate ZMP-based feasible CoM trajectories purely in an open-loop strategy, the choice of feedback control can then be freely designated.

Considering real-time sensory feedback control, impedance control seems to be the general approach.

By inserting a floating-base inverse dynamics term as feedforward torques, it is possible to achieve sufficient joint position tracking with relatively lower gains, so that the locomotion becomes more compliant [5] [6]. The main difficulty of this method is based on inverse dynamics; if one or more legs are not in contact with the ground for a specified instant, some undesired kicking motions occur. Such undesired motions become much more significant in robots with comparatively large mass and inertia, such as HyQ [7]. To remedy this issue, we simply removed inverse dynamics and run the system on a high gain PD servo controller. In parallel to this loop, we have an additional admittance controller which decreases the joint stiffness when force error appears. Such force error creates an additional position feedback term to perturb position reference, in a way to comply with the force constraint. As soon as the external force/disturbance causing the force error vanishes, the joint turns back to its initial stiffness for position tracking. This trade-off is modulated by a set of admittance coefficients.

In the light of the above discussion, the contributions of this paper can be grouped in two directions: a) We propose a CoP-based CoM trajectory generator, in which initial conditions are uniquely set for physically-viable motion characteristics. The algorithm is purely computational without any inner feedback loop, easy-to-apply and analytical. b) An admittance control scheme is adopted to achieve active compliance control for quadrupedal locomotion, in which environmental interactions are compliantly handled. The controller does not need inverse dynamics and can introduce active compliance with the help of an additional feedback loop that is based on joint force error.

The paper is organized as follows. We briefly introduce our experimental platform in section II. Section III is devoted to explain the derivation of trajectory generator. The controller algorithm is explained in detail in section IV. Experimental results are thoroughly evaluated in section V. Finally, the paper is concluded in section VI by stating final discussions.

II. HyQ Quadruped

Our platform, HyQ (see Fig. 1), is a fully torque-controlled hydraulically and electrically actuated quadruped robot comparable in size to a goat ($\sim$75*kg*), e.g. an *Alpine ibex*. It has been designed and built *in-house* and it uses a combination of hydraulic cylinders and electric motors for the actuation of its 12 joints [7]. HyQ is capable of highly dynamic locomotion as hydraulic actuation allows the handling of large impact forces, high bandwidth control, high power-to-weight ratio and superior robustness.

Each leg has three degrees of freedom (*DoFs*), two in the hip (abduction/adduction (hip_{aa}) and flexion/extension (hip_{fe})) and one in the knee (flexion/extension ($knee_{fe}$)). The hip_{aa} joints are actuated by electric motors while all the hip_{fe} and the $knee_{fe}$ joints are hydraulically actuated. All of the quadruped's joints are equipped with high resolution encoders and load cells, which allow a smooth control of both position and torque. Overall the robot weighs $\sim$75*kg*, it is 1*m* long and 0.5*m* wide and stands 1*m* high with the legs fully stretched. The system is controlled by a Pentium PC104 running real-time patched Linux (*Xenomai*) and is capable of reaching a 1*kHz* control frequency.

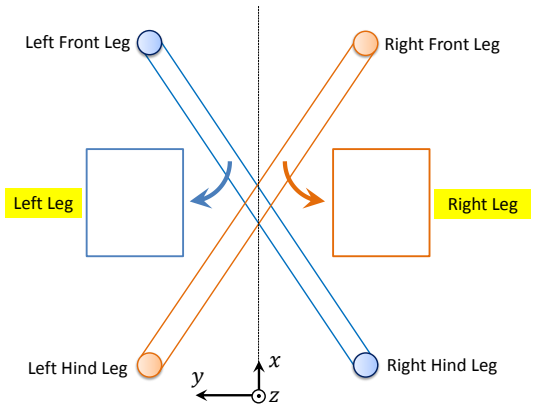

Fig. 2. Corresponding feet positions of quadruped trot-walking, in equivalent planar biped model.

III. Trajectory Generation

In quadrupedal trot-walking, robot feet are diagonally paired as shown in Fig. 2. While the robot is in motion, these paired legs move simultaneously. In a continuous trot-walking, there are three phases: i) Left front, right hind stance phase; ii) 4 legs stance phase, iii) Right front, left hind stance phase. Since there is no 3 legs stance phase, we can create an analogy for an equivalent planar biped (EPB) model, considering the particular trot-walking motion (see Fig. 2). That being said, left foot of biped is the middle point of left front - right hind couple. In a similar manner, right foot of biped is the middle point of right front - left hind couple. Aforementioned phases then can be interpreted as i) left foot single support phase, ii) double support phase, iii) right foot single support phase. To simplify the concept, we will use this model in generating quadrupedal trot-walking trajectories.

In addition, our experimental experiences indicate that quadruped trot-waking can be achieved without any lateral motion, i.e., the robot does not have to sway towards left or right since it is equally balanced on the diagonally paired legs. Keeping this in mind, we only focus on sagittal plane motion generation in this study.

A. CoM Trajectory

As previously stated, we make use of CoP criterion to generate dynamically balanced CoM trajectories. When considering point mass model, ZMP and CoP (Center of Pressure) coincides; therefore, we use the term CoP instead of ZMP. Let us begin with analyzing CoP equations for a trot-walking scenario with constant CoM height [8].

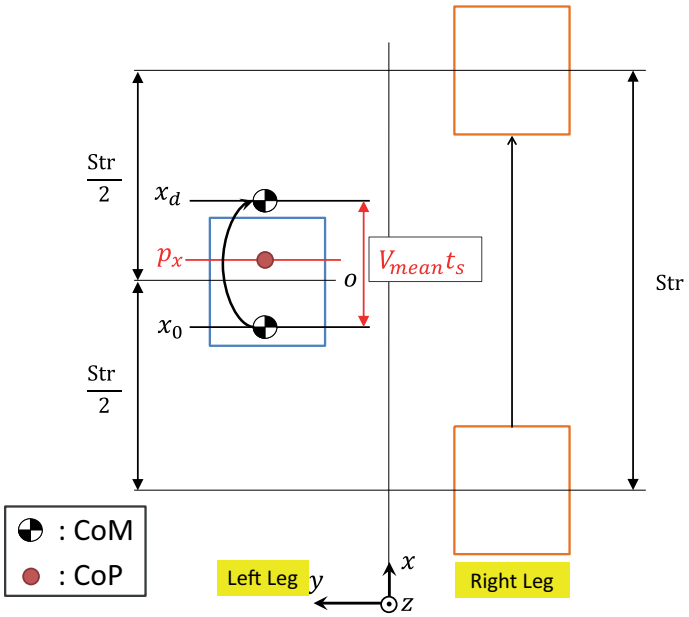

Fig. 3. EPB model during single support phase.

$$X_{cop} = x - z\frac{\ddot{x}}{g} \tag{1}$$

In (1), (x, z) are horizontal and vertical CoM position, g is gravitational acceleration, and X_{cop} is x-axis CoP. Depending on the X_{cop} input, we may solve (1) analytically. It should be noted that (1) is derived via total angular momentum rate change (total moment), in which CoP acts like an inertial frame. Thus, X_{cop} can be selected either a step or a ramp input. Any other alternatives create non-zero acceleration for CoP and violate Newton's law; therefore, it may lead to theoretically wrong applications [1].

In our trajectory generator, X_{cop} is constant during single support phases ($X_{cop} = p_x$), and linearly increasing during double support phases ($X_{cop} = K_x t + p_x$).

1) CoM Trajectory During Single Support Phase: During single support phases, we consider a constant X_{cop} input; ($X_{cop} = p_x$). In this case, (1) can be analytically solved as follows;

$$x = (x_0 - p_x)\cosh\omega\tau + \frac{\dot{x}_0}{\omega}\sinh\omega\tau + p_x \tag{2}$$

$$\omega = \sqrt{\frac{g}{z}}, \quad \tau = t - t_0, \tag{3}$$

in which x_0, $\dot{x}_0$, ω, t, t_0 are initial CoM position, initial CoM velocity, equivalent pendulum natural frequency, time variable and initial time, respectively. CoM velocity and acceleration functions can also be obtained via differentiation, as in the following.

$$\dot{x} = \omega\,(x_0 - p_x)\sinh\omega\tau + \dot{x}_0\cosh\omega\tau \tag{4}$$

$$\ddot{x} = \omega^2\,(x_0 - p_x)\cosh\omega\tau + \omega\dot{x}_0\sinh\omega\tau. \tag{5}$$

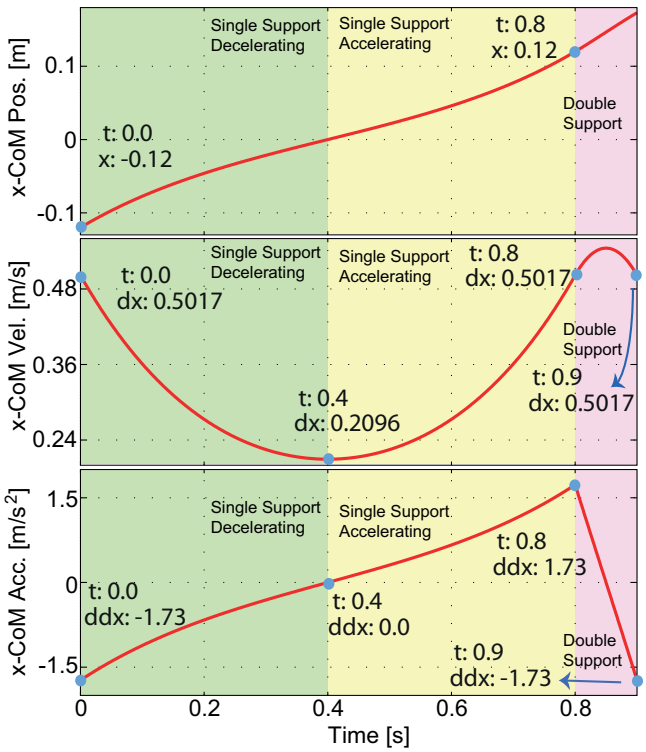

Fig. 4. Feasible CoM motion during a single (0.0-0.8 secs.) and a double (0.8-0.9 secs.) support phase periods.

In order to obtain a physically viable CoM trajectory, a single support phase should be composed of two equal deceleration and acceleration phases. A numeric example is illustrated in Fig. 4, in sequence with a double support period. As a crucial feature of a viable CoP-based CoM trajectory, initial CoM velocity should be equal to terminal velocity. It also makes initial acceleration and terminal acceleration equal in amplitude with opposite signs. This property can be guaranteed by setting initial conditions in accordance with desired motion parameters. Since the nature of ZMP equations are hyperbolic, any single support phase with unbalanced deceleration/acceleration periods will accumulate exponential increases in CoM position and produce physically unrealizable trajectories.

Using the above-mentioned symmetry property, we may realize that velocity position has a minimum at the middle point of a single support period, i.e., when $t = t_m = t_0 + T_s/2$, where T_s is single support time period.

$$\ddot{x}|_{t=t_m} = \omega^2(x_0 - p_x)\cosh(\frac{wT_s}{2}) + \omega\dot{x}_0\sinh(\frac{wT_s}{2}) = 0 \tag{6}$$

Utilizing (6), one can yield $\dot{x}_0$ as below.

$$\dot{x}_0 = \omega(p_x - x_0)\coth(\frac{wT_s}{2}) \tag{7}$$

During a single support phase, we may substitute an mean forward velocity, v_{mean}. Considering this parameter, the terminal position, x_d, can be computed (Fig. 3).

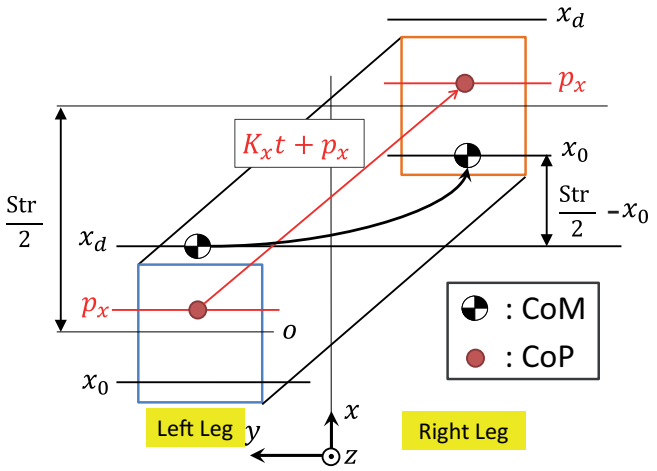

Fig. 5. EPB model during a double support phase.

$$x_d = v_{mean} T_s + x_0. \tag{8}$$

Terminal position x_d can also be computed by using (2), when $t = t_e = t_0 + T_s$:

$$x|_{t=t_e} = x_d = (x_0 - p_x)\cosh(\omega T_s) + \frac{\dot{x}_0}{\omega}\sinh(\omega T_s) + p_x. \tag{9}$$

Inserting (7) into (9), hyperbolic terms cancel each other out ($\cosh \omega T_s - \sinh \omega T_s \coth 0.5\omega T_s) = -1$), and following equation is yielded.

$$x_d + x_0 = 2p_x \tag{10}$$

Combining (8) and (10), x_0 and x_d terms are calculated in the following manner.

$$x_0 = p_x - \frac{v_{mean} T_s}{2}; \quad x_d = 2p_x - x_0 \tag{11}$$

For a given set of single support period (T_s), target mean velocity (v_{mean}), and constant X_{cop} input, we may sequentially calculate initial conditions (x_0,$\dot{x}_0$) and via (11) and (7). x_d can also be calculated beforehand for prior verification.

2) CoM Trajectory During Double Support Phase: The main task during double support phase is to carry CoP from the preceding support foot to the proceeding support foot, without any discontinuity as displayed in Fig. 5. Furthermore, at the end of a double support phase, CoM velocity and acceleration values must be coincided with initial CoM velocity and acceleration values for the next single support phase; so that we can connect the sequential phases seamlessly. Referring to Fig. 5, we may solve differential equation (1) for a linearly increasing X_{cop} input ($X_{cop} = K_x t + p_x$). Its first time derivative also provides us velocity function as well.

$$x = (x'_0 - p_x)\cosh \omega\tau' + \frac{\dot{x}'_0 - K_x}{\omega}\sinh \omega\tau' + K_x\tau' + p_x \tag{12}$$

$$\dot{x} = \omega(x'_0 - p_x)\sinh \omega\tau' + (\dot{x}'_0 - K_x)\cosh \omega\tau' + K_x \tag{13}$$

In (12) and (13), (x'_0,$\dot{x}'_0$) are initial position and velocity terms for a double support phase. They are equal to terminal position and velocity of the preceding single support phase, as indicated in (14). K_x is a slope value for X_{cop}. $\tau' = t'_0 + t$ stands for shifted time variable with initial time for double support period, namely t'_0. For a seamless connection, terminal velocity of a double support phase should be equal to initial velocity of the following single support phase when $t = t_{ed} = t'_0 + T_d$.

$$x'_0 = x_d; \quad \dot{x}'_0 = \dot{x}_0; \tag{14}$$

$$\dot{x}|_{t=t_{ed}} = \dot{x}_0 = \omega(x_d - p_x)\sinh \omega T_d + (\dot{x}_0 - K_x)\cosh \omega T_d + K_x. \tag{15}$$

Assessing (15), K_x and stride lenght, Str, may be obtained as follows.

$$K_x = \dot{x}_0 + \omega(x_d - p_x)\coth \frac{\omega T_d}{2}; \tag{16}$$

$$Str = 2T_d K_x \tag{17}$$

3) Trajectory Generation Algorithm: One can easily generate CoM trajectories via the following algorithm.

1) Assign single/double support periods (T_s,T_d), constant CoM height (z), mean target velocity (v_{mean}) and constant ZMP input for a single support period (p_x).
2) Sequentially compute ω, x_0, x_d, $\dot{x}_0$, K_x and Str via (3), (11), (7), (16), and (17).
3) For single support phases, use (2) to generate CoM trajectory. For double support phases, use (12) to generate CoM trajectory.

This algorithm makes sure that sequential single and support phases are tied seamlessly, both for X_{cop} and corresponding x, $\dot{x}$, $\ddot{x}$ trajectories, through the exploitation of symmetry properties as explained above. Please refer to sample code for MATLAB[1] for further examination.

B. Swing Leg Trajectories

Swing leg trajectories are constructed using polynomials by considering the Stride length (Str) and maximum vertical swing foot clearance. In generating swing leg trajectories, both initial and terminal velocity and acceleration terms are set to zero to make sure that swing foot does arrive to the floor in a motionless state, so that impact forces may be reduced.

[1]https://dl.dropbox.com/u/21136797/UgurluIROS13SampleCode.zip

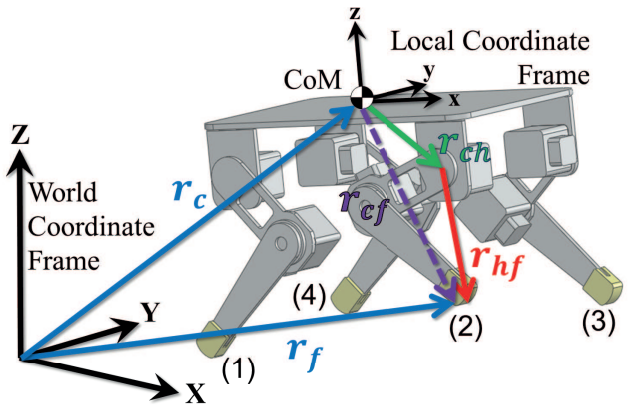

Fig. 6. Vectorial quantities on the quadruped. Legs are also numbered.

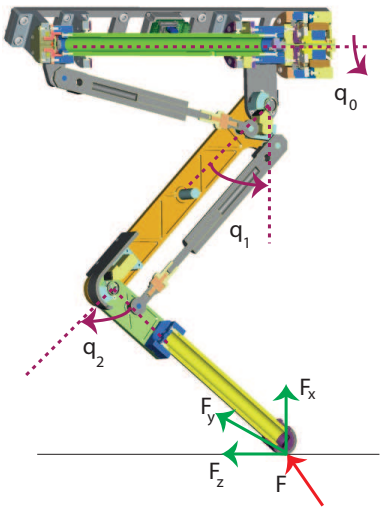

Fig. 7. Ground reaction force at a foot tip.

C. Joint Motion Computation

In order to realize CoM and swing leg trajectories, we are in need of computing the position vector that is defined between the respective hip and foot position for each leg. If we refer to Fig. 6, the associated vectorial operation for the 2^{nd} leg can be expressed as below.

$$\mathbf{p_{hf}} = R_b^T(\mathbf{p_f} - \mathbf{p_c}) - \mathbf{p_{ch}} \tag{18}$$

In (18), $\mathbf{p_f}$ is a foot position vector with respect to world frame, $\mathbf{p_c}$ is the CoM position vector with respect to world frame, R_b is upper torso orientation that is composed of three possible rotations about pitch (θ) and (ψ) roll and yaw (γ) axes, $\mathbf{p_{ch}}$ is a position vector between CoM and the respective hip. After calculating $\mathbf{p_{hf}}$ vector for each leg, joint angles can be analytically computed, via the utilization of a 3-DoF inverse kinematics scheme.

In summary, we define 18 inputs for the joint motion generation task; CoM position (3 elements), each foot positions (3×4 elements), and torso orientation (3 elements). In total, we define 18 inputs, whereas the robot has only 12-DoF. This may seem to be an underactuation issue; however, diagonal pairs move simultaneously during trotting motion. That being said, 6 inputs are not imposed to system, making our calculation to have a unique solution; 12 inputs for 12 DoF.

IV. Active Compliance Control

A. Referential Force Calculation

In the proposed active compliance controller, we simultaneously process position and force references. Therefore, one needs to define joint force references in accordance with the target trajectory. To attain this goal, we primarily define 3 equations using vertical forces.

$$Mg = \sum_{i=1}^{4} F_{zrefi}; \tag{19}$$

$$X_{copref} = \frac{\sum_{i=1}^{4} (F_{zrefi}\ r_{cfxrefi})}{\sum_{i=1}^{4} F_{zrefi}}; \tag{20}$$

$$Y_{copref} = \frac{\sum_{i=1}^{4} (F_{zrefi}\ r_{cfyrefi})}{\sum_{i=1}^{4} F_{zrefi}}; \tag{21}$$

In (19)-(21), M is total mass, g is gravitational acceleration, F_{zrefi} is i^{th} foot vertical force reference, X_{copref} and Y_{copref} are CoP references, $r_{cfxrefi}$ and $r_{cfyrefi}$ refer to referential displacement between CoM and i^{th} foot, through x-axis and y-axis. In (19), we equalized total force to the robot weight, since there is no vertical acceleration during trot-walking. Eqs. (20) and (21) allow us to utilize force distribution; usually both CoP references are kept at zero.

In addition to (19)-(21), we need to define one more condition, since we have 4 legs. This can be obtained from zero-yawing motion. However, this condition is related to horizontal (F_{xrefi}) and lateral (F_{yrefi}) forces. Even though the robot has point contact feet, the total ground reaction force is decomposed depending on the foot plate orientation (see Fig. 7). That being said, (F_{xrefi}) and (F_{yrefi}) can be expressed in terms of F_{zrefi}, so that we may obtain the 4^{th} condition.

$$F_{xrefi} = F_{zrefi} \tan(q_1 + q_2) \sec q_0; \tag{22}$$

$$F_{yrefi} = -F_{zrefi} \tan q_0; \tag{23}$$

$$\begin{aligned} & F_{yref1} r_{cfxref1} - F_{xref1} r_{cfyref1} + F_{yref2} r_{cfxref2} \\ - & F_{xref2} r_{cfyref2} + F_{yref3} r_{cfxref3} - F_{xref3} r_{cfyref3} \\ + & F_{yref4} r_{cfxref4} - F_{xref4} r_{cfyref4} = 0. \end{aligned} \tag{24}$$

Referring to Fig. 7, q_0, q_1, and q_2 respectively denote roll axis hip (hip_{aa}), pitch axis hip (hip_{fe}), and pitch axis knee ($knee_{fe}$) joints. Combining (22)-(24) would give us the 4^{th} equation. In addition to (19)-(21), all these equations enable

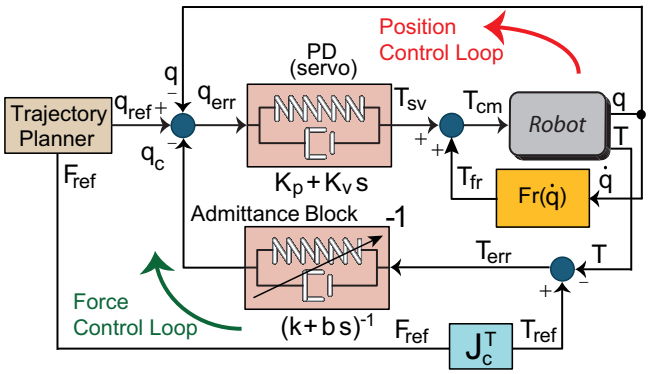

Fig. 8. Active compliance control in the servo loop.

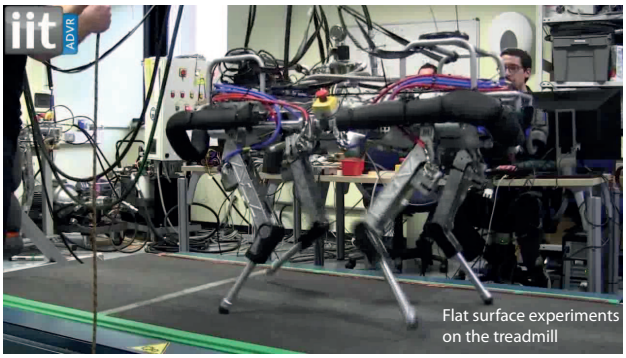

Fig. 9. A snapshot from level surface trot-walking experiments, conducted on a treadmill.

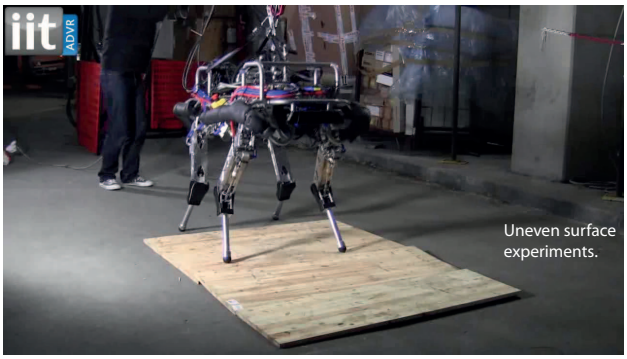

Fig. 10. A snapshot from uneven surface trot-walking experiments, conducted outdoor. Three wooden plates with 3 [cm] thicknesses are placed on the floor, in a way that the robot gradually steps over.

us to compute vertical force references for each foot, during double support phase where all 4 feet are on the ground. During single support phases, we have only 2 feet on the ground; thus, we only utilize (19) and (20) to compute vertical force references. Swing foot force references are naturally set to zero. Once we have vertical force reference for the i^{th} foot (F_{zrefi}), we can compute horizontal and lateral components (F_{xrefi},F_{yrefi}) using (22) and (23). Having computed all the 3-D referential force vector for the i^{th} leg, we make use of Jacobi transpose to obtain referential joint forces.

B. Joint Force Feedback: An Antagonistic Approach

Fig. 8 depicts the controller that is implemented in the proposed method. In this figure, q_{ref}, q and q_c stand for referential joint angles, actual joint angles, and output of admittance block. T_{ref}, T_{sv}, T_{fr}, T_{cm}, T, and T_{err} respectively denote referential, PD output, friction compensation, command, actual (measured) and error torques. K_p and K_v refer to PD gains, k and b show admittance block coefficients.

Primarily, trajectory generator provides referential joint angles which are inserted to PD servo controllers. On top of this scheme, we added a force control loop in which joint force errors are processed via an admittance block to compute corresponding joint displacements, namely. By perturbing joint reference for about q_c degrees, joint force feedback is provided. In this scheme, basically PD controller makes sure that position tracking is achieved. However, within the presence of force errors, q_ref is updated via the secondary feedback (q_{ref} : $q_{ref} - q_c$) to comply with force constraints, by decreasing the joint stiffness. Once, the external effect that causes force error (disturbances, ground impact, etc.) q_c term vanishes and joint turns back to its initial stiffness. The trade-off between position and force tracking is adjusted via admittance block parameters. In other words, force control loop and position control loop works in an antagonistic configuration; the joint becomes compliant when necessary to handle force errors. If there is no force error, it prioritizes joint tracking. In addition, friction compensation is added to enhance mechanical performance.

In [5], [6], [11], a low gain PD with floating-based inverse dynamics based approach is adopted. While useful in its own right, this approach has two shortcomings: i) Compliance control is not associated with force feedback. Regardless of the force error, the joint always keeps the same stiffness/softness. ii) Inverse dynamics terms require the stance foot information. Regrettably, the current HyQ version has only joint torque sensors and force estimation could be prone to noises. Due to this fact, the robot may perform unexpected kicking motions. Our proposed controller, on the other hand, associates force feedback with an active compliance control strategy. It can also be used without inverse dynamics. In this matter, we would like to highlight the fact that additional inverse dynamics surely escalate our controller performance as well and we will add this option when HyQ is equipped with force sensors at the feet points.

V. Experiment Results

In order to validate the proposed trajectory generator and controller, we conducted two different experiments. i) Trot-walking on a level surface (treadmill), ii) Trot-walking on uneven surface (outdoor). In both experiments, trot-walking parameters are set as the same. Single and double support time periods are 0.28 [s] and 0.14 [s], constant CoM height is 0.68 [m], constant ZMP input is 0 [m], maximum vertical swing foot clearance is 10 [cm]. Target mean velocity is set as 1.26 [km/h] for level surface experiments and 0.36 [km/h] for uneven surface experiments. Fig. 9 and Fig. 10 display

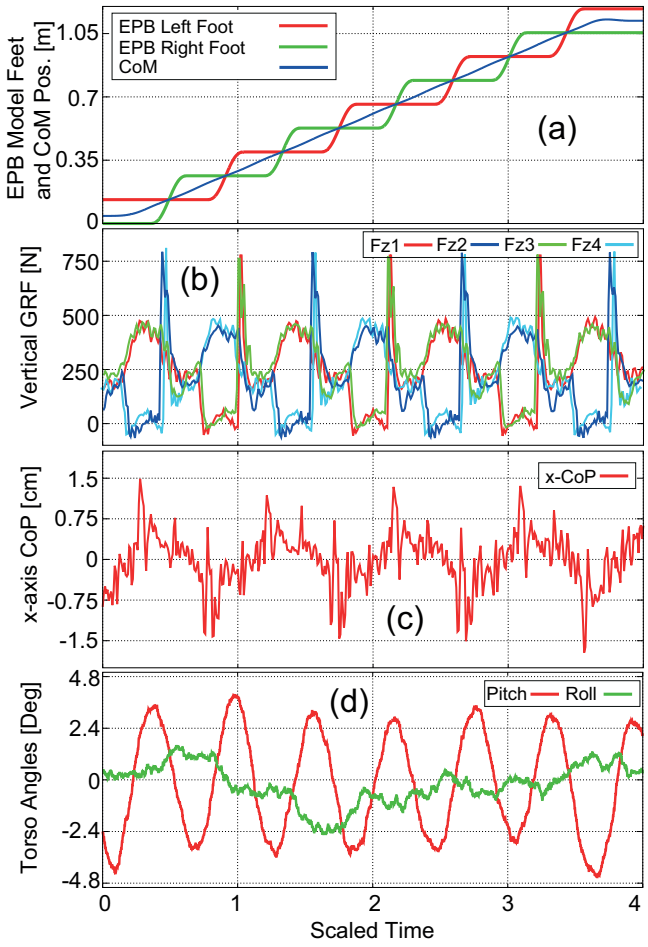

Fig. 11. Level surface experiment results.

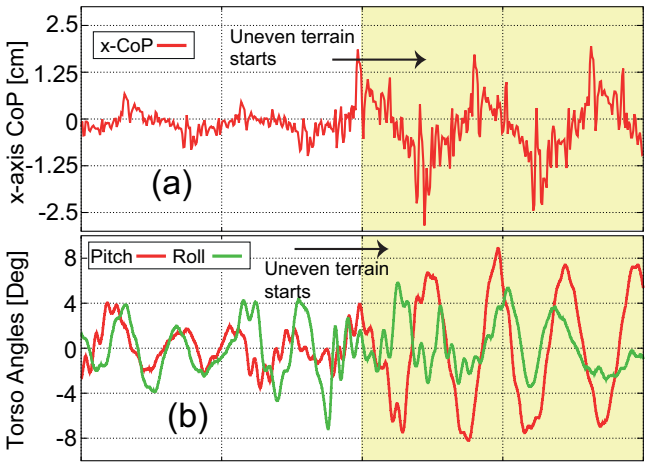

Fig. 12. Uneven surface experiment results.

a couple of snapshots from the experiments, which can be viewed in the multimedia attachment of this paper.

A. Level Surface Trot-Walking Experiments

To begin with, we conducted level surface experiments on a treadmill (see Fig. 9). Fig. 11 displays the results of this experiment.

Referential feet displacement and CoM trajectory with respect to world frame is shown in Fig. 11(a). Feet positions are plotted with EPB model in mind (see Fig. 2) to provide a clearer view. Judging by this figure, we may claim that the proposed trajectory generator is able to synthesize smooth and continuous motion patterns, which are seamlessly connected through single and double support phases. Trajectories are also observed to be smooth and continuous in velocity and acceleration levels, but not plotted due to page restrictions. Note that the plot displays only 8 steps; however, real experiments include much longer experimentation periods.

Vertical ground reaction forces (GRFs) are estimated using joint torque and position sensors and depicted for each leg in Fig. 11(b). We may see that diagonally paired legs move simultaneously as their swing and stance phases seem to be overlapped. In this figure, it is possible to observe two benefits that are gained through the utilization of active compliance controller: i) Reaction forces are equally distributed to each leg, so that the robot can exhibit dynamically balanced locomotion characteristics. ii) Touch-down impacts are relatively low for a $\sim 75[kg]$ quadruped robot with stiff-by-nature actuation configuration, resulting in a more compliant and efficient trot-walking locomotion.

x-axis CoP response variation with respect to local body frame is displayed in Fig. 11(c). Scrutinizing this figure, X_{cop} varies between ± 1.5 [cm]. Even though the support polygon is a diagonal line for trot-walking locomotion, the actual support polygon can be stretched ± 3 [cm] due to non-zero foot plate surface. Therefore, any local CoP variation between ± 3 [cm] margins are considered to point out dynamically balanced locomotion. This result does vary within this range, indicating that the level surface trot-walking experiments are conducted in a dynamically equilibrated manner.

Torso angle variations are also collected via an IMU module and plotted in Fig. 11(d). In this figure, we may see that pitch and roll torso angle variations are respectively kept within ± 4.8 [Deg] and ± 2.4 [Deg] ranges. This result gives us a clue about the trot-walking performance; thanks to active compliance controller, the robot is not prone to undesired torso angle fluctuations which may be caused by inequal reaction force distribution to legs or large ground reaction force impacts. Based on this result, it may be claimed that the controller potentially escalates dynamic trot-walking performance.

B. Uneven Surface Trot-Walking Experiments

Upon the successful completion of level surface experiments, HyQ is moved outside of the lab and trot-walking experiments are conducted on an uneven terrain (see Fig. 10). For this purpose, 3 wooden plates with 3 [cm] thickness are gradually placed on top of each other, both creating unevenness and moderate slope. Trajectory generator is not modified; however, only target forward velocity is reduced to eliminate the risk any possible damage to the robot. Fig.

12 points out the results from this experiment. Due to page restrictions, only CoP and torso angle variations are included.

x-axis CoP variations with respect to local body frame is illustrated in Fig. 12(a). This measurement primarily follows a variation trend between ± 0.7 [cm]. As the robot starts walking on the uneven terrain, it starts varying with a larger amplitude which is observed to be $+1.7 \sim -2.5$ [cm]. Even though the variation amplitude enlarges, it is still within the feasible support polygon range which is ± 3 [cm]. Observing this outcome, we may conclude that the robot is able to walk on the specified uneven terrain with dynamically balanced trot-walking locomotion cycles.

Torso angle variations are included in Fig. 12(b). Initially, our measurements show a variation trend around ± 4 [Deg]. Once the robot starts trot-walking on the uneven terrain, pitch angle variation amplitude gets bigger and becomes ± 8 [Deg]. Roll angle variation also show a slight increase. Increases in torso angle variations are expected, this is a price to be paid while the robot is performing outdoor trot-walking on an irregular surface. Nevertheless, this result did not influence the overall dynamic balance and the robot successfully completed its task.

VI. Final Discussions

In this paper, we proposed an analytical trajectory generation algorithm and an active compliance controller, which are combined in dynamic trot-walking task for an hydraulically actuated, stiff-by-nature ($\sim 75kg$) quadruped robot. To validate the proposed schemes, two types of dynamic trot-walking experiments are conducted: i) Level surface trot-walking on a treadmill, ii) uneven surface trot-walking. In both experiments, the robot is able to complete the tasks with dynamic, repetitive, continuous, and dynamically balanced trot-walking locomotion characteristics.

The trajectory generator is confirmed to be able to synthesize reliable trot-walking locomotion cycles, which are smooth, continuous and correspond to feasible CoP inputs. The algorithm is easy-to-implement and only requires a couple of parameters to be tuned. Comparing to other trajectory generators, the number of free parameters are relatively low and it does not dictate any inner feedback loop.

The active compliance controller is also observed to be efficient in handling environmental interaction while the robot is trot-walking. It enabled us to obtain equally distributed reaction forces and relatively lower touch-down impacts. These properties directly related to overall trot-walking performance, and thus, allow us to escalate the locomotion efficiency.

In our future work, we will seek ways to include CPGs into CoP (or, in other words, ZMP)-based trajectory generation algorithm to synthesize a method which possess advantages of both approaches. The active compliance controller scheme will be supported with full body floating-base inverse dynamics computation once the HyQ is equipped with force sensors at the feet tips. In addition, our team is implementing an overall state estimation based on sensor fusion. As soon as these improvements are realized, we conduct highly rough terrain locomotion experiments on HyQ.

Acknowledgements

The authors would like to thank Kana Kotaka, Jesus Ortiz, Jake Goldsmith, Marco Frigero, Michele Focchi, Thiago Boaventura, Stephane Bazeille, Bilal Rehman, Hamidreza Saghir, Hamza Khan, Isil Cakir Ugurlu, and our team of technicians for their kind assistance and support throughout the completion of this work.

References

[1] T. Sugihara, and Y. Nakamura, "Boundary condition relaxation method for stepwise pedipulation planning of biped robots", in *IEEE Trans. on Robotics*, vol. 25, no. 3, 2009, pp. 658-669.

[2] M. Hutter, M. A. Hoepflinger, C. Gehring, M. Bloesch, C. D. Remy, and R. Siegwart, "Hybrid operational space control for compliant legged systems", in *Proc. Robotics: Science and Systems*, Sydney, Australia, 2012.

[3] H. Kimura, Y. Fukuoka, and A. H. Cohen, "Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts", in *Int. Journal of Robotics Research*, vol.26, no.5, pp. 475-490, 2007.

[4] M. Raibert, K. Blankespoor, G. Nelson, R. Playtor, and the Big-Dog Team, "Bigdog, the rough-terrain quadruped robot", in *Proc.* 17^{th} *World Cong. The Int. Fed. Automatic Control*, Seoul, Korea, 2008, pp. 10822-10825.

[5] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal, "Learning, planning and control for quadruped locomotion over challenging terrain", in *Int. Journal of Robotics Research*, vol. 30, no. 2, 2011, pp. 236-258.

[6] T. Boaventura, C. Semini, J. Buchli, M. Frigero, M. Focchi, and D. G. Caldwell, "Dynamic torque control of a hydraulic quadruped robot", in *Proc. IEEE Conf. on Robotics and Automation*, St. Paul, US, 2012, pp. 1889-1894.

[7] C. Semini, N. G. Tsagarakis, E. Guglielmino, M. Focchi, F. Cannella, and D. G. Caldwell, "Design of HyQ - A hydraulically and electrically actuated quadruped robot", in *Proc. Ins. Mech. Eng, Part I: Journal of Systems and Control Engineers*, vol. 225, no. 6, 2011, pp. 831-849.

[8] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point", in *Proc. IEEE Conf. on Robotics and Automation*, Taipei, Taiwan, 2003, pp. 1620-1626.

[9] K. Byl, A. Shkolnik, S. Prentice, N. Roy, and R. Tedrake, in "Reliable dynamic motions for a stiff quadruped", *Springer Tracks in Advanced Robotics*, vol. 54, 2009, pp. 319-328.

[10] S. Rutishauser, A. Sprowitz, L. Righetti, and A. J. Ijspeert, "Passive compliant quadruped robot using central pattern generators for locomotion control", in *Proc. IEEE Conf. on Biomedical Robotics and Biomechatronics*, Scottsdale, US, 2008, pp. 710-715.

[11] V. Barasuol, J. Buchli, C. Semini, M. Frigero, E. R. De Pieri, and D. G. Caldwell, "A reactive controller framework for quadrupedal locomotion on challenging terrain", in *Proc. IEEE Conf. on Robotics and Automation*, Karlsruhe, Germany, 2013, to be presented.

[12] C. Maufroy, H. Kimura, and K. Takase, "Integration of posture and rythmic motion controls in quadrupedal dynamic walking using phase modulations based on leg loading/unloading", in *Autonomous Robots*, vol.28, no.3, pp. 331-353, 2010.

[13] F. L. Moro, A Sprowitz, A. Tuleu, M. Vespignani, N. G. Tsagarakis, A. J. Ijspeert, and D. G. Caldwell "Horse-like walking, trotting, and galloping derived from kinematic motion primitives (kMPs) and their application to walk/trot transitions in a compliant quadruped robot", in *Biological Cybernetics*, Forthcoming article, 2013.

[14] T. Koolen, T. de Boer, J. R. Rebula, A. Goswami, and J. E. Pratt, "Capturability-based analysis and control of legged locomotion, Part 1: Theory and application to three simple gait models", in *Int. Journal of Robotics Research*, vol.31, no.9, pp. 1094-1113, 2012.

[15] T. Akbas, S. E. Eskimez, S. Ozel, O. K. Adak, K. C. Fidan, and K. Erbatur, "Zero moment point based pace reference generation for quadruped robots via preview control", in *Proc. IEEE Works. on Advanced Motion Control*, Sarajevo, Bosnia&Herzegovina, 2012, pp. 1-7.