A Novel Dynamic Locomotion Control Method for Quadruped Robots Running on Rough Terrains

CHAO DING, LELAI ZHOU, (Member, IEEE), YIBIN LI, (Member, IEEE), AND XUEWEN RONG, (Member, IEEE)
Center for Robotics, School of Control Science and Engineering, Shandong University, Jinan 250061, China
Engineering Research Center of Intelligent Unmanned System, Ministry of Education, Jinan 250061, China
Corresponding author: Lelai Zhou (zhoulelai@sdu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFC0806505, and in part by the National Natural Science Foundation of China under Grant 91948201, Grant 61973191, and Grant U1613223.

I. INTRODUCTION

Quadruped robots have been widely studied due to the inspiration that quadruped animals in nature have significant advantages of flexibility, terrain adaptability, and kinematic performances [1]. Quadruped robots are mainly designed for the uses of rescue, detection, transportation, and other dangerous tasks in unstructured natural environments. It is expected that they can work in extremely rugged areas where wheeled vehicle cannot reach.

The most famous control architecture for legged robot is the three-part theory proposed by Marc Raibert [2]. The control system decomposes the problem into three weak coupling parts, the hopping amplitude, the forward speed, and the body attitude. The most famous quadruped robot BigDog [3], which is the pioneer of hydraulic actuated bionic legged robots, is based on this algorithm. The follow-up robot LS3, designed by Boston Dynamics, is also widely recognized to adopt the three-part theory. Although it is similar in scale and weight with the robot studied in this work, no detailed information has been published.

From the perspective of bionics, the walking motions of quadruped animals are generated by a central pattern generator (CPG) [4], [5]. The CPG receives sensory signals of adjustments from the brain via reflexes. Kimura [6] proposed a neural system composed of the CPG, responses, and reflexes mechanism. The method was applied to the robot Tekken2 to walk on the ground. Barasuol [7] proposed a CPG-based trajectory generator for the quadruped robot HYQ designed by Italian Institute of Technology to adapt to challenging terrains and lateral pushes. Zhang [8] combined the CPG and the fuzzy control strategy on a quadruped model, but only showed the effectiveness on the flat terrain. Wang [9] constructed a parallel multi-module controller based on CPG and realized the basic walking motions of the robot named PMQ. Wang [10] combined the CPG and workspace trajectory generator to achieve robot locomotion on flat terrain with multi-gait, but the parameters needed to be tuned manually.
for different gaits. Liu [11] proposed a multi-layer CPG model to control the quadruped robot walking on 10° slopes. Generally, the CPG-based approaches often require a complicated non-linear oscillator to build the controller, and have strong dependency on parameters.

An intuitionistic method for the controlling of legged robots is the Virtual Model Control (VMC) proposed by Pratt [12]. The VMC applies appropriate forces to the robot via the virtual spring-damping model mounted on the bearing point. Gehring [13] distributed forces for the standing legs of Starl-ETH by VMC and converted the solution into an optimization problem. The robot realized multi-gait locomotion and 5 cm high obstacles adaptation. Koco [14] embedded the virtual model in the single leg of the quadruped robot to realize compliance control but had limited ability of body control. Tian [15] took both the virtual model control of each single leg and the non-linearity of actuators into consideration, but has not extended it to the control of the whole body. Xie [16] controlled both the swing phase and the standing phase based on the VMC, which only showed the effectiveness on flat terrain. Zhang [17] realized the dynamic trot gait control of Scalf (the previous generation of ScaFL-III studied in this work) in simulations, but it abandoned the control of the lateral degree of freedom. Generally, the VMC of the robot trunk relies on the precise estimation of the virtual stiffness and virtual damping coefficients, which is also the experience obtained when we try to apply VMC to the previous generation of SCalf-III. It is sensitive to these two parameters.

Some researchers introduce the zero moment point (ZMP) method into the control of quadruped robots, which is commonly used in the locomotion of bipeds. Ugurlu [18] generated a dynamically equilibrated center of mass (CoM) trajectory for the robots RoboCat-1 and HyQ [19], which could walk on a rocky terrain. Zhang [20] designed a ZMP-based trot gait for the quadruped robot with twisting trunk whereas lacked the control of trunk attitudes. Kalakrishnan [21] optimized the body trajectory based on the ZMP for a quadruped robot LittleDog [22], combining with the terrain template learning algorithm to realize rough terrain locomotion. The robot needed to be trained off-line and the terrain map was generated ahead of each experiment. The ZMP-based approaches are usually used in the static gaits and lack the capacity of disturbance resistance.

Some novel control architectures appear recently. Hwangbo [23] deployed the reinforcement learning method on the platform ANYmal. The MIT Cheetah 3 [24] showed superior dynamic locomotion performances with multiple gaits through model predictive control (MPC), but the approach needed global state estimations, which usually caused accumulative errors.

To overcome these shortcomings, a novel trot gait control architecture is proposed in this work for quadruped robots running on multi-terrains. The system is composed of the VMC based swing phase controller and the MPC based controller for standing phase, which are connected by the state machine. The swing leg trajectory cluster is generated to reach the desired landing point with the capability of external disturbance tolerance. Swing leg forces are calculated by a virtual spring-damping model on the foot to follow the desired trajectory. The dynamic model of the trunk is built in the standing-leg-related reference coordinate without the requirement of global state estimation, which commonly introduces accumulative errors to the system. The state prediction is realized by the state recursion based on the dynamic model. The force distribution of standing legs is converted into a quadratic optimization problem solved by the optimizer Gurobi. The gravity compensation and the attitude adjustment on uneven terrains are considered in the dynamic model. Simulation results show the effectiveness of the proposed method. The quadruped robot can run on uneven terrains and has robustness to lateral pushes.

The paper is organized as follows. Section II briefly introduces the robot and the simulation model. Section III builds the dynamic model for the trunk during the standing phase and obtains the recurrence relation of the robot states. Section IV distributes forces for standing legs via predictive quadratic optimization. Control strategies for the swing legs are given in Section V. Section VI integrates the sub-controllers by the state machine. Simulations are carried out to show the effectiveness in Section VII. Section VIII discusses the advantages of this work. Section IX gives the conclusion.

**II. SYSTEM OVERVIEW**

Our research team released the third generation of ScaLF series quadruped robots named ScaLF-III in 2018, which was designed to achieve transportation tasks under unstructured environments. The robot consists of the body and four legs with three hydraulic actuated joints in each, one in the hip for adduction and abduction and two in the hip and knee for flexion and extension, respectively. Comparing with the precious generations, ScaLF-III is larger in scale and have better locomotion performances. This work focuses on the dynamic locomotion on rough terrains for ScaLF-III based on its simulation model, as shown in Fig. 1. The main parameters of the robot and the model are listed in Table 1. Mechanically, the mass of the robot is designed to concentrate mainly on the trunk and distribute uniformly. Light-weight actuators and materials are used to construct the legs so that the leg inertia can be neglected in most cases comparing to the trunk.
III. MODELING OF STANDING PHASE

For the quadruped robot running in trot gait with the duty factor of 0.5, there are two diagonal legs standing on the ground at any moment. The standing legs provide forces and torques to drive the robot to perform the desired actions.

A. DYNAMICS

Since the mass of the leg is pretty small comparing to the trunk, and the mass of the trunk distributes uniformly, the quadruped robot can be regarded as a rigid body that takes the geometric center of the trunk as the center of gravity (CoG). It only suffers the contact forces and the gravitational force. The dynamics of the robot can be derived based on the Newton-Euler approach. For the convenience of the observations of the robot states, the origin of the motion reference coordinate {\(\mathcal{O}\)} is defined as the midpoint of the two standing feet as shown in Fig. 2. The \(x_r-y_r\) plane coincides with the ground. The \(x_r\)-axis and \(y_r\)-axis are parallel to the projections of \(x_b\)-axis and \(y_b\)-axis onto the ground. \({\mathcal{O}}\) is the base frame.

The dynamic model of the quadruped robot in the motion reference coordinate can be expressed by

\[
\begin{align*}
\sum_{i=1}^{n} \left( r_f \times F_i \right) &= \dot{r}_T + \omega \times (\dot{r}_T) \\
\frac{d}{dt} \dot{r}_F &= \frac{1}{m} \sum_{i=1}^{n} F_i + \dot{r}_G \\
\frac{d}{dt} \Theta &= \omega \\
\frac{d}{dt} \dot{r}_P &= \dot{r}_F
\end{align*}
\]

(1)

where \(r_f\) is the lever arm vector of the contact force \(F_i = [f_{i,x}, f_{i,y}, f_{i,z}]^T\) for the \(i\)-th standing leg. \(\Theta = [\gamma, \theta, \varphi]^T\) is the attitude angle of the robot. \(\omega\) is the angular velocity. \(r_T\) is the inertia tensor of the robot body. \(r_P = [x_r, y_r, z_r]^T\) is the position of the center of gravity which is assumed to locate at the center of the trunk. \(r_G\) is a compensation term corresponding to the gravitational force. \(i\) is the identifier of the standing leg, and \(n\) is the amount of the standing legs. For the trot gait, \(n = 2\).

In the case that the state of the robot accurately follows the desired state trajectory, the body maintains parallel to the reference ground. There are only translations but no rotations between the base frame and the motion reference coordinate.

The position of the CoG in the reference frame can be obtained by

\[
\dot{r}_P = -\frac{1}{2} (b_{P_{st,1}} + b_{P_{st,2}})
\]

(2)

where \(b_{P_{st,i}}\) is the position of the standing leg with respect to the base frame and can be calculated through the forward leg kinematics.

The moment arms of contact forces in the reference frame and the base frame are equivalent

\[
\dot{r}_i = b_{r_i} = b_{P_{st,i}}.
\]

(3)

B. SIMPLIFICATION AND TERRAIN ADAPTATION

When the quadruped robot is running on the flat ground, the roll and pitch angles in the world coordinate frame are expected to be zero. This strategy needs to be adjusted when the robot encounters uneven terrains, such as slopes and stairs. In this work, the \(x_b-y_b\) plane is regulated to be parallel to the ground surface instead of the horizontal plane in these cases. The concepts of pseudo roll and pitch angles are introduced to the dynamic model as shown in Fig. 3.

The pseudo pitch angle \(\theta\) is defined as the angle between the \(x_b\) and \(x_r\) axes in the sagittal plane, which is observed as the angle between the robot body and the connecting line of the standing feet in the \(x_r-z_r\) plane. Thus, the pseudo pitch angle used in \(\Theta\) is calculated as

\[
\theta = \arctan \frac{b_{z_f} - b_{z_h}}{b_{x_f} - b_{x_h}}
\]

(4)

---

**TABLE 1. Main parameters of the quadruped robot and the simulation model.**

| Specifications       | Symbols/Units | Values |
|----------------------|---------------|--------|
| Body                 | mass          | \(m_{\text{model}}\) K | 275 |
|                     | length        | \(L_b\) m | 1.36 |
|                     | width         | \(W_b\) m | 0.57 |
| Leg                  | mass          | \(m_{\text{leg}}\) K | 15  |
| Upper link length    | \(L_0\) m    | 0.045  |
| Middle link length   | \(L_1\) m    | 0.451  |
| Lower link length    | \(L_2\) m    | 0.405  |

---

**FIGURE 2.** Dynamic model of the body in the standing phase.

**FIGURE 3.** The definition of the pseudo pitch angle.

---

C. Ding et al.: Novel Dynamic Locomotion Control Method for Quadruped Robots Running on Rough Terrains

VOLUME 8, 2020 150437
Similarly, the pseudo roll angle is

\[ \gamma = \arctan \frac{b_{x_f} - b_{z_f}}{b_y + b_r} \]  

where \([b_{x_f}, b_{y_f}, b_{z_f}]^T\) and \([b_{x_h}, b_{y_h}, b_{z_h}]^T\) are the positions of the front and hind standing legs in the base frame. \([b_{x_f}, b_{y_f}, b_{z_f}]^T\) and \([b_{x_h}, b_{y_h}, b_{z_h}]^T\) correspond to the positions of the left and right legs. In order to maintain the accurate global heading direction of the robot, the yaw angle in the dynamics is the same as that in the inertial frame, which is measured by the Inertial Measurement Unit (IMU).

To get the maximum stability margin, the quadruped robot usually runs along the longitudinal direction when climbing slopes. Based on the normalized energy stability margin (NESM) [25], the projection of the CoG onto the ground should keep close to the center of the standing legs, as shown in Fig. 4. The adjustment of the CoG is

\[ \Delta = r_z \tan \phi \]

where \(\phi\) is the slope angle of the terrain, which can be measured by the IMU assuming that the robot precisely follows the desired state trajectory (\(\theta \approx 0\) and \(\gamma \approx 0\)).

On uneven terrains, the gravity not only contributes in the z direction of the robot, but also have effective component along the x-axis. The compensation term to balance the influence of the gravity in the dynamics should be

\[ rG = [g \sin \phi \ 0 \ -g \cos \phi]^T \]

where \(g\) is the gravitational acceleration.

In the case that the robot follows the desired trajectory exactly, the term \(\omega \times (rI \omega)\) does not contribute significantly to the Euler dynamic equation comparing to the term \(rI \dot{\omega}\) since the angular velocities are very small. It can be neglected. The Euler dynamics is approximated to be

\[ \frac{d}{dt} \omega \approx rI^{-1} \sum_{i=1}^{n} (r_i x \times rF_i) = rI^{-1} \sum_{i=1}^{n} [r_i x] x rF_i \]

where \([r_i x]\) is the skew-symmetric matrix of \(r_i x\) to transfer the cross products of matrices into dot products.

The mass of the quadruped robot is designed to distribute uniformly in the cuboid-shaped trunk. The inertial tensor in the robot in the base frame is roughly a diagonal matrix \(bI = \text{diag}(b_{I_{xx}}, b_{I_{yy}}, b_{I_{zz}})\). The inertial tensor in the reference frame can be derived as

\[ rI = bR bI bR^T + m[(rP^T rP)A_{3 \times 3} - rP rP^T] \]

where \(A_{3 \times 3}\) is the 3 \(\times\) 3 identity matrix. \(bR\) is the rotation matrix from the base frame to the reference frame which is an expression of the pseudo attitude angles. Based on the intrinsic relationship between the two frames and the zero-desired pseudo attitude angles, the rotations between them are negligible, and thus

\[ rI \approx m[(rP^T rP)A_{3 \times 3} - rP rP^T]. \]

C. STATE RECURSION

With the approximations and the terrain adaptation strategies above, the dynamic model of the robot with two diagonal standing legs can be expressed in the state space form

\[
\begin{align*}
\frac{d}{dt} & \begin{bmatrix} \Theta \\ \omega \\ rP \\ r\dot{P} \end{bmatrix} = \\
& \begin{bmatrix}
0_{3 \times 3} & A_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
\end{bmatrix} \\
& \bigg[ \\
& \begin{bmatrix}
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
A_{3 \times 3/m} & A_{3 \times 3/m} & A_{3 \times 3/m} & A_{3 \times 3/m} \\
0_{3 \times 1} & 0_{3 \times 1} & 0_{3 \times 1} & 0_{3 \times 1} \\
\end{bmatrix} \\
& \begin{bmatrix} rF_1 \\ rF_2 \end{bmatrix} \\
& + \\
& \begin{bmatrix}
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
\end{bmatrix} \\
& \bigg].
\end{align*}
\]

Mark the state and the force input of the robot as \(X(t) = [\Theta, \omega, rP, r\dot{P}]^T\) and \(U(t) = [rF_1, rF_2]^T\), respectively. The state equation can be written as

\[
\dot{X}(t) = AX(t) + B(t)U(t) + G(t)
\]

where \(A \in \mathbb{R}_{12 \times 12}\) and \(B(t) \in \mathbb{R}_{12 \times 6}\) and are the system matrix and input matrix corresponding to those in (11). \(G(t) \in \mathbb{R}_{12 \times 1}\) is the gravity compensation matrix. Strictly speaking, (11) is a non-linear time-varying system since the inertial tensor and the level arm in the input matrix \(B(t)\) vary with time. The state equation can be discretized by introducing the state transition matrix after being converted into the standard linear form. For simplification, in the case that the sampling period is small enough, the discretization can be approximated as

\[
X(k + 1) - X(k) = A(k)X(k) + B(k)U(k) + G(k)
\]

where \(\Delta t\) is the sampling period, and \(k\) means the \(k\)-th sampling instant of the system.

Thus, the robot states at the future moments can be derived from the current one. The recursive relation is

\[
X(k + 1) = \bar{A}(k)X(k) + \bar{B}(k)U(k) + \bar{G}(k)
\]

where

\[
\bar{A}(k) = A(k)\Delta t + A_{12 \times 12}
\]

\[
=[
\begin{bmatrix}
A_{3 \times 3} & A_{3 \times 3}\Delta t & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & A_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & A_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & A_{3 \times 3} \\
\end{bmatrix}
\]
\[ \mathbf{B}(k) = \mathbf{B}(k) \Delta t \]

\[ = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ r^{T}I^{-1}r_{1} \Delta t & r^{T}I^{-1}r_{2} \Delta t \\ 0_{3 \times 3} & 0_{3 \times 3} \\ \Lambda_{3 \times 3} \Delta t/m & \Lambda_{3 \times 3} \Delta t/m \end{bmatrix} \]  \tag{16}

\[ \mathbf{G}(k) = \mathbf{G}(k) \Delta t \]

\[ = \begin{bmatrix} 0_{1 \times 3} & 0_{1 \times 3} & 0_{1 \times 3} & r^{T}G \Delta t \end{bmatrix}^{T}. \]  \tag{17}

Note that \( \mathbf{A}(k) \) is a constant matrix that is only decided by the sampling period. \( \mathbf{B}(k) = \mathbf{B}(k; r_{1}, r_{2}) \) varies at each point of the robot state trajectory. It is defined by the feedback and desired positions of the CoG and the standing feet. \( \mathbf{G}(k) = \mathbf{G}(\phi) \) is determined by the terrain angle which is measured by the IMU on the robot.

**IV. PREDICTIVE FORCE DISTRIBUTION OF THE STANDING LEG WITH QUADRUPED OPTIMIZATION**

The human-robot interaction is carried out by a wireless remote controller. The robot receives only three parameters from the operator, the linear velocities along \( x \) and \( y \) directions and the yaw angular velocity. The desired robot states at each point of the prediction horizon are determined as follows.

- The desired pseudo roll and pitch angles, the pseudo roll and pitch angular velocities, and the desired linear velocity along \( z \) direction are always set to be zero.
- The desired yaw angle, \( x \) position, and \( y \) position are calculated by the current state and the integration of the corresponding desired velocities.
- The desired \( z \) position is the standing height of the robot which is set as a constant value in the controller.

In the control of the quadruped robot, both the current state and the future states are expected to follow the desired state trajectory. The concept of predictive control can be borrowed. The contact forces of the standing legs are distributed to minimize the following equation

\[ W(U) = \sum_{k=0}^{l-1} \left[ \|X(k+1) - X_{\text{ref}}(k+1)\|_{Q(k)}^{2} + \|U(k)\|_{R(k)}^{2} \right] \]  \tag{18}

where \( X_{\text{ref}}(k+1) \) is the desired state of the \((k+1)\)-th sampling along the desired state trajectory. \( l \) is called the length of the prediction horizon. \( Q(k) \in \mathbb{R}_{12 \times 12} \) is a diagonal positive definite matrix for the weight of each element of the state vector. \( R(k) \in \mathbb{R}_{6 \times 6} \) is the diagonal positive definite matrix to limit the magnitude of the inputs, also the consumption of the system. The minimization of \( W(U) \) can get a balance between the performance and the cost of the quadruped robot. The operator \( \|Y\|_{H} \) is the ellipsoidal norm of \( Y \), which is defined as

\[ \|Y\|_{H} = \sqrt{H^{T}YH}. \]  \tag{19}

Practically, the standing foot suffers only thrust forces but no pull forces from the ground that is

\[ r_{f_{i,z}} \geq 0. \]  \tag{20}

To avoid the foot sliding, the three dimensions of the contact force \( rF_{i} \) are constrained by the friction cone with the frictional coefficient \( \mu \). With the constraints above, the distribution of the contact forces for the standing legs can be converted into a convex quadratic optimization problem described by

\[ \begin{aligned}
\min_{\substack{i=0 \ldots l-1}} & \; \left[ (ax - b)^{T} Q(k)(ax - b) + x^{T} R(k)x \right] \\
\text{s.t.} & \; -\mu r_{f_{i,z}} \leq r_{f_{i,x}} \leq \mu r_{f_{i,z}} \\
& \; -\mu r_{f_{i,z}} \leq r_{f_{i,y}} \leq \mu r_{f_{i,z}} \\
\end{aligned} \]  \tag{21}

where

\[ \begin{aligned}
x & = U(k) \\
a & = \mathbf{B}(k) \\
b & = X_{\text{ref}}(k+1) - \mathbf{A}(k)X(k) - \mathbf{G}(k). \\
\end{aligned} \]  \tag{22}

The contact forces are also restricted by the threshold \( rF_{i,max} \), which is caused by the limitations of the joint torques and the current leg configuration. This is not listed in (21) for briefness.

The first six elements in the solution of (21) are the optimal forces that should be exerted to the standing legs relative to the motion reference coordinate. They are approximately equal to the forces in the base frame, when the robot precisely follows the desired state trajectory. Then, the joint torques can be obtained by introducing the Jacobian.

**V. MODELING OF THE SWING PHASE**

The swing legs find the optimal and safe regions for the quadruped robot to land the feet while matching the operation demands. In this section, the trajectory cluster of the swing foot is generated based on its landing point. The force distribution is realized by the VMC.

**A. LANDING POINT OF THE SWING FEET**

For the quadruped robot running at the desired speed \( \dot{x}_{d} \), the swing legs should move across the distance of \( \dot{x}_{d} T_{sw}/2 \) during the whole period of each swing phase. Here, \( T_{sw} \) is the running time of each swing phase. However, the actual speed \( \dot{x} \) hardly matches the desired value, while unexpected external disturbances also bring uncertainties to the control of the speed. These errors can be adjusted by a gain compensation with the coefficient \( k_{1} \). The landing point of the swing foot on the horizontal plane is derived as

\[ x_{sw,f} = \frac{\dot{x}_{d} T_{sw}}{2} + k_{1}(\dot{x} - \dot{x}_{d}) \]  \tag{23}

\[ y_{sw,f} = \frac{\dot{y}_{d} T_{sw}}{2} + k_{j}(\dot{y} - \dot{y}_{d}). \]  \tag{24}

According to our previous researches, the quadruped robot can get better performances with the gain coefficient of \( \sqrt{H_{b}/g} \). \( H_{b} \) is the desired standing height of the robot. In the \( z \)-axis, the foot should return to the position of \( -H_{b} \) at the end of every swing phase to maintain stable height of the trunk, which is

\[ z_{sw,f} = -H_{b}. \]  \tag{25}
B. TRAJECTORY OF THE SWING FEET

Generally, the trajectory of the swing feet should match the principles and purposes as follows.
- Reduce the impulse and the slide between the foot and the ground.
- Maximize the highest position in the z direction to overcome obstacles.
- No jump points in the position curve and the velocity curve.
- Appropriate mechanisms to cope with the possible earlier/later touch-down.

Since the landing point position varies, the trajectory of the swing foot is not a single curve but a curve cluster. In this work, the trajectory cluster is generated in the incremental form

\[
\begin{align*}
    x_{sw}(k+1) &= x_{sw}(k) + \frac{x_{sw,f} + \Delta x_{sw} - x_{sw}(k)}{T_{sw} - t} \Delta t_{sw} \\
y_{sw}(k+1) &= y_{sw}(k) + \frac{y_{sw,f} + \Delta y_{sw} - y_{sw}(k)}{T_{sw} - t} \Delta t_{sw}
\end{align*}
\]

where \(x_{sw}(k+1)\) and \(y_{sw}(k+1)\) are the desired positions during the next servo cycle. \(x_{sw}(k)\) and \(y_{sw}(k)\) are the current feedback positions. \(t\) is the passed running time of the current swing phase and \(\Delta t_{sw}\) is the servo interval of the swing leg. \(\Delta x_{sw}\) is a compensation term corresponding to the terrain adaptation, and is set to be the opposite value of \(\Delta\) in (6)

\[
\Delta x_{sw} = -\Delta.
\]

\(\Delta y_{sw}\) is the compensation term in y-axis similar with \(\Delta x_{sw}\), and is set to zero in this work.

The trajectory along z-axis is designed simply as a sine curve

\[
z_{sw}(t) = -H_b + h_{sw} \sin\left(\frac{t}{T_{sw}}\pi\right)
\]

where \(h_{sw}\) is the desired step height.

C. FORCE GENERATION OF THE SWING LEG BASED ON VMC

The swing foot should be controlled to follow the trajectory defined by (26) (27) and (29). The concept of the VMC is introduced to this work. A three-dimensional hypothetical spring-damping model is mounted at the foot between the current and desired positions, as shown in Fig. 5. The force that should be applied to the swing leg is calculated as

\[
F_{sw} = k_p(P_{sw} - P_{sw,d}) + k_p(\dot{P}_{sw} - \dot{P}_{sw,d})
\]

where \(P_{sw}\) and \(\dot{P}_{sw}\) are the feedback of the position and the velocity. \(P_{sw,d}\) and \(\dot{P}_{sw,d}\) are the desired position and the desired velocity. \(k_p\) and \(k_p\) are the gains corresponding to the virtual stiffness and damping.

VI. CONTROL IMPLEMENTATION

The subcontrollers proposed above are harmonized by a state machine to constitute the control architecture for the quadruped robot, as shown in Fig. 6.

Three commands are sent to the robot via the operator, the desired \(x\) velocity, the desired \(y\) velocity, and the desired steering angular velocity. Forces of the standing legs are generated to maintain the desired robot states, while the forces of the swing legs are obtained by following the desired foot trajectory. Leg forces are then converted into joint torques with the Jacobian.

This work focuses on the trot gait with the duty factor of 0.5. At any moment, there are two diagonal legs in standing phase, while the other leg pair is in the swing phase. The running time of the two phases is \(T_{sw} = T_{sf} = T/2\), \(T\) is the period of one gait cycle.

The switch between the flying and standing phases and the unambiguous judgement of the phases of each leg are of great importance to quadruped robots. Since the legs are controlled by different strategies during different leg phases, the inaccurate judgement leads to wrong actions and instability, even damages to the robot. The state machine in this work is based on the fusion of the time information and the contact force. The phase of a swing leg is switched into the standing phase once it satisfied that

\[
t \geq \alpha T \land f_c \geq F_{th}
\]

where \(f_c\) is the contact force calculated by the feedback joint torques and the Jacobian. \(F_{th}\) is a force threshold to verify that the foot has contacts with the ground. The coefficient \(\alpha\)
is set to be 0.5 under ideal conditions for the trot gait, and can also be tuned in a small scale for more superior performance. The phase switch strategies for the other legs are similar since they have the fixed phase difference.

VII. SIMULATION AND RESULTS
Simulations are executed in the simulator Webots on the quadruped model built referring to the mechanical structures and physical parameters of Scalf-III. The robot model is simulated to run on various terrains and suffer external disturbances to verify the effectiveness of the proposed method. The system uses three types of sensors, the position and force sensors in each joint and the IMU on the trunk. The resolution of the position sensors is set to be 0.05 mm. The resolution of the attitude angle measurement is 0.01°. The values of all the sensors are updated at the frequency of 250 Hz, which is also the running frequency of the controller. All of these configurations are within the capacity of the off-the-shelf hardware products.

A. OMNI-DIRECTIONAL LOCOMOTION ON FLAT TERRAIN
The simulation is firstly executed on flat terrains to test the omni-direction locomotion of the quadruped robot. The robot receives stochastic mixed human commands of $x$-velocity, $y$-velocity, and yaw angular velocity from the remote operator. There is no other external human interference to the robot besides these three kinds of orders. The desired values in Fig. 7(a) ~ 7(c) compose the command sequence. The robot is commanded to conduct multi-directional linear motions, curve motions, and steering at different periods of the entire simulation.

The simulation results indicate that the states of the robot precisely follow the desired state trajectory. The coupling between the different states is extremely weak. The robot maintains upright attitudes along the whole mission with no more than 0.003 rad errors, as shown in Fig. 7(d). There are two periods of small deviations between the desired and the feedback $y$ velocity at about $t \in (5\,s \sim 10\,s)$ and $t \in (34\,s \sim 40\,s)$. These are caused by the centrifugal force due to the coexistence of the longitudinal movement and the steering motion. In the human-robot interaction mode, the controller does not pursue the high-precision control of the global positions. The centrifugal force needs no composition under this circumstance. The highest velocities in $x$-axis and $y$-axis stably succeeded are 2.1 m/s and 0.7 m/s. Lots of parameter tuning experiences are obtained through the simulation such as that, the larger value of the weight of $r_y$ (corresponding to the fourth diagonal element of $Q_k$) leads to smaller overshoot and longer settling time, and the smaller value of the weight of $r_z$ (corresponding to the twelfth diagonal element of $Q_k$) leads to body height oscillations. Other parameters in the weight matrices have the similar effects.

Remark: This work focuses on the artificial manipulation of the quadruped robot. The velocities analyzed in this work are observed in the local coordinate system. They approximate to the observations in the global frame, if there is no sliding between the foot and the ground. In fact, the foot sliding rarely happens due to the constraints of the friction cone in the quadratic optimization problem. It provides the extensibility to use the proposed method for global navigation tasks. Under the global circumstance, the accumulative errors need to be compensated, since the integration of velocities is used to obtain the robot position.

B. ADAPTATION TO THE SLOP
The quadruped robot successfully runs over 10°, 20°, and 30° hill-like slops in the simulation. Fig. 8 is the snapshots of the simulation when the robot climbs up and down the 20° slop. The results of tramping over the 20° slop are shown and compared to the currently used method in Fig. 9.
The robot runs over the 20° slop, (a) the feedback roll and pitch angles of the body, (b) the feedback roll and pitch angles using the SLIP-based method, (c) the feedback pseudo pitch angle, (d) the desired and feedback x velocity, (e) the feedback forces of the standing legs, (f) the CoG position with respect to \{\mathbf{r}_O\}.

The robot starts at the flat ground when the desired x velocity of 0.3 m/s is transported via the operator. The robot then climbs up the 20° slop. After arriving at the platform at the top of the slop, the robot runs down the −20° slop. The up and down process is coherent with no other external assistants to the robot except the velocity command at the very start. The sampling frequency is reduced to 10 Hz for clearness of the curves. As shown in Fig. 9(a) and Fig. 9(c), the pitch angle of the body measured by the IMU matches the gradient of the slop, while the feedback pseudo pitch angle maintains close to zero, which means the robot perfectly implements the proposed terrain adaptation strategy for the attitude angles. The roll angle and the x velocity also follow the desired values well. The errors of the states become slightly larger during the time of \(t \in (10.5 \text{ s} \sim 12.5 \text{ s})\) and \(t \in (27 \text{ s} \sim 29 \text{ s})\). They are caused by the foot-ground impacts, since the robot is in the junction region of the flat terrain and the slop. The errors are acceptable since they do not introduce instability to the robot. Fig. 9(f) shows the CoG position with respect to the motion reference frame \('\mathbf{O}'\), which is also illustrated by the auxiliary lines in Fig. 8. The adjustment amount of the CoG position coincides with the tendency of the pitch angle, since they have the relationship of (6). The jumps in the CoG position curve indicate the phase switches of the legs. For comparison, we repeat the simulation using the spring loaded inverted pendulum (SLIP) based control method, which is currently used by Scalf-II and Scalf-III [26], as shown in Fig. 9(b). It takes the robot more time to climb down the slop, since the robot is resisted in the junction area of the slop and the ground. Comparing to the SLIP-based method, the fluctuation amplitude of the roll angle reduces about 80% with the method proposed in this work.

The locomotion stability is evaluated based on the concept of stability margin when the robot climbs the slops. The stability margin in this work is quantified as the distance on the horizontal plane between the projection of CoG and the projection of the midpoint of standing legs, as

\[
S_{\text{loc}} = -\frac{b_x f_x + b_h f_h}{2} \cos \phi - \frac{b_z f_x + b_z f_h}{2} \sin \phi. \tag{32}
\]

The robot is commanded to climb slops with the expected speed \(\dot{x}_d = 0.3 \text{ m/s}\) and the step frequency \(f_{\text{step}} = 1.5 \text{ Hz}\). The five-number summary [27] of the stability margin is shown in Fig. 10. The closer to zero of \(S_{\text{loc}}\) indicates the larger stability margin of the robot. The theoretical range of \(S_{\text{loc}}\) is \([-\frac{\dot{x}_d}{f_{\text{step}}}, \frac{\dot{x}_d}{f_{\text{step}}}]\). Simulation results show that the locomotion stability is remarkably improved with the proposed terrain
adaptation strategy, since the value of $S_{loc}$ is closer to zero and the outliers are significantly decreased. The robot fails to climb up the 30° slope without the terrain adaptation method.

C. CLIMBING UP AND DOWN STAIRS

The ability of running over stairs is tested in this section. The robot is commanded to climb up and down continuous stairs in the dynamic trot gait without any priori knowledge about the terrain. The stairs are composed of six steps and each step is 8 cm high and 45 cm wide. Fig. 11 shows six typical moments during the simulation.

Fig. 12 shows the performance of the robot. During the entire simulation, the feet of the robot fall down from the edge of the steps for 17 times at $t = \{4.4 \text{ s}, 4.8 \text{ s}, 7.7 \text{ s}, 9.5 \text{ s}, 9.8 \text{ s}, 11.4 \text{ s}, 11.8 \text{ s}, 12.2 \text{ s}, 13.4 \text{ s}, 13.8 \text{ s}, 14.1 \text{ s}, 27.1 \text{ s}, 28.3 \text{ s}, 29.7 \text{ s}, 31.0 \text{ s}, 32.2 \text{ s}, 33.8 \text{ s}\}$. The impulses and the sudden changes in the altitude angle curves, the velocity curves and the standing height curve are mainly caused by these falling down. The worst violent case happens at the time of $t \in (7.5 \text{ s} \sim 12.5 \text{ s})$, during which the front legs slip down from the cliff of the edge over and over again. The unexpected foot falling causes deviation of the roll angle of the body from the desired value. The robot perfectly handles these emergencies by lateral adjustments on the stairs, which correspond to the suddenly increased values of the $y$ velocity in Fig. 12(c).

The robot also conquers a single step of 10 cm high with dynamic trot gait locomotion. The results are shown in Fig. 13. Falling of the standing legs also happens for three times when the front legs reach the edge of the step ($t \in (1.5 \text{ s} \sim 3 \text{ s})$) and is well managed. No parameters in the controller are changed comparing with those used on flat terrains. The quadruped robot achieves step and stairs climbing blindly. This work realizes continuous stairs climbing of Scalf series in the dynamic gait for the first time. Although Scalf-II can achieve single step climbing, the robot needs time to recover the leg length after each step since the method [26] is based on the variable stiffness and variable leg length (VSVL).

D. LATERAL IMPACT RECOVERY

The ability of recovery from external disturbances is an important performance evaluation index for quadruped robots. In this simulation, the robot is knocked by a 100 Kg weight bob with the pendulum length of 0.7 m. The impulse exerted to the robot is 370 Kg·m/s. The adjustments and results are shown in Fig. 14 and Fig. 15. It takes two initiative steps for the robot to recover from the lateral push. Different from the currently used method on Scalf-III [26], the lateral impact does not cause obvious tilts of the roll angle, which means that the method in this work has better performance on attitude control of the quadruped robot. The impulse mainly causes the unanticipated lateral velocity, which leads to lateral adjustments according to (24). The results show that the robot has strong robustness to external perturbations with the proposed controller.
E. ENERGY EFFICIENCY AT DIFFERENT SPEED

The energy efficiency of the proposed approach is considered in this section comparing with the SLIP based method [26] at different running speed, as shown in Table 2.

| Method   | $U_{cost}/J \cdot m^{-1}$ | Speed/m \cdot s^{-1} |
|----------|-----------------------------|-----------------------|
| MPC+VMC  | 1386.23                     | 0.3                   |
|          | 1080.13                     | 0.6                   |
|          | 1181.64                     | 1.0                   |
| SLIP [26]| 2202.11                     | 0.3                   |
|          | 1500.99                     | 0.6                   |
|          | 1959.40                     | 1.0                   |

TABLE 2. Energy consumption at different running speed.

where $j$ is the identifier for the joints, $\tau_j$ is the torque of the joint and $\dot{q}_j$ is the joint angular velocity. The term in the denominator is the running distance during the running time $T_{des}$. The major purpose of this work is to improve the terrain adaptability of quadruped robots, meanwhile, the higher energy efficiency is achieved. The results support that the proposed method reduced the energy cost by $29\% \sim 40\%$ comparing to the SLIP based approach. Note, the simulations are conducted with the step frequency of 1.5 Hz and the step height of 0.15 m, and the running distance is measured by an external global observer.

VIII. DISCUSSION

This section depicts the advantages of this work comparing to the previous locomotion control methods for the three generations of SCalf robot series. The trot gait controller for the first generation of SCalf was based on position control and did not take the foot contact force impacts into account, which often caused mechanical damages to the robot. The static walk gait [28] was designed for the robot to adapt to stairs and uneven terrains whereas the walking speed was too slow. The method could not be extended to be used in dynamic gaits. Hua [29] proposed a passive compliance method for hydraulic actuators but the method was still not applied to the robot prototype. The VMC based method [17] achieved the whole-body modeling and control of SCalf, but the approach deserted the control in the lateral direction since the matrix used to calculate the forces was singular. The currently used compliant control method for trot gait on SCalf-II and SCalf-III was based on the SLIP model [26]. This method achieved the compliance of each leg based on the trajectory generation, which was based on the positions inherently. It was difficult for all of the four legs to achieve harmonious compliance performances due to the differences of the actuator characteristics. It had capability for the single step but not continuous stairs.

The control approach proposed in this paper achieves omni-directional locomotion of SCalf-III on various rough terrains with robustness to lateral impacts. The modelling during the standing phase realizes full-scale control of the whole body. The compliance character of the foot-ground interaction is achieved inherently since the forces are distributed based on the optimal body performances. The quadruped
A dynamic trot gait locomotion control approach for the quadruped robot was proposed in this work. The force planning for swing legs accommodated to the external disturbances. The force distribution for standing legs was based on the predictive quadratic optimization involving the adjustment strategies on rough terrains. The proposed method was not sensitive to parameters and needed no global state observations, which helped to avoid accumulative errors comparing with the traditional MPC based methods. The quadruped robot achieved omni-directional locomotion, 30° slope climbing, continuous 8 cm high stairs climbing, and push recovery in simulations.

This method has not been implemented on the prototype of ScafIII since it is sensitive to the accuracy of the CoG position. In simulations, when the lateral offset of the CoG is 5 cm, the undesired lateral velocity of 0.2 m/s occurs, which needs to be compensated. Our ongoing researches are devoted to detect the CoG position for quadruped robots in the dynamic gait. The proposed method will be implemented on the prototype in the future with the compensation of CoG offsets.

REFERENCES

[1] A. M. Wilson, J. C. Lowe, K. Roskill, P. E. Hudson, K. A. Golabeck, and J. W. McNutt. “Locomotion dynamics of hunting in wild cheetahs,” Nature, vol. 498, no. 7453, pp. 185–189, Jun. 2013.

[2] M. H. Raibert and E. R. Tello. Legged Robots That Balance. Cambridge, MA: MIT Press, 1986.

[3] D. Wooden, M. Malchano, K. Blankspoor, A. Howard, A. A. Rizzi, and M. Raibert. “Autonomous navigation for BigDog,” in Proc. IEEE Int. Conf. Robot. Autom., May 2010, pp. 4736–4741.

[4] P. Ngamkajornwiwat, J. Honchhanthanakul, P. Peerakitikul, and P. Manoonpong. “Bio-inspired adaptive locomotion control system for online adaptation of a walking robot on complex Terrains,” IEEE Access, vol. 6, pp. 8042–8065, 2018.

[5] L. Miniti, M. Frasca, N. Yoshimura, and Y. Koike. “Versatile locomotion control of a hexapod robot using a hierarchical network of nonlinear oscillator circuits.” IEEE Access, vol. 6, pp. 4804–4823, 2018.

[6] H. Kimura, Y. Fukuoka, and A. H. Cohen. “Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts,” Int. J. Robot. Res., vol. 26, no. 5, pp. 475–490, May 2007.

[7] V. Barasuol, J. Buchli, C. Semini, M. Frigerio, E. P. De Pieri, and D. G. Caldwell. “A reactive controller framework for quadrupedal locomotion on challenging Terrain,” in Proc. IEEE Int. Conf. Robot. Autom., May 2013, pp. 2554–2561.

[8] D. Zhang, J. Wu, Q. Zhu, R. Xiaoyi, Z. Zhang, Y. Zha, and J. Xu. “Fuzzy control method based on central pattern generator for QuadrupedRobot,” in Proc. Chin. Autom. Congr. (CAC), Nov. 2019, pp. 861–866.

[9] B. Wang, Z. Wan, C. Zhou, J. Wu, Y. Qiu, and Z. Gao. “A multi-module controller for walking quadruped robots,” J. Bionic Eng., vol. 16, no. 2, pp. 252–263, Mar. 2019.

[10] M. Zhang, Z. Tang, B. Chen, and J. Zhang. “Locomotion control for quadruped robot based on central pattern generators,” in Proc. 35th Chin. Control Conf. (CCC), Jul. 2016, pp. 6335–6339.

[11] C. Liu, L. Xia, C. Zhang, and Q. Chen. “Multi-layered CPG for adaptive walking of quadruped robots,” J. Bionic Eng., vol. 15, no. 2, pp. 341–355, Mar. 2018.

[12] J. Pratt, C. M. Chew, A. Torres, P. Dilworth, and G. Pratt. “Virtual model control: An intuitive approach for bipedal locomotion,” Int. J. Robot. Res., vol. 20, no. 2, pp. 129–143, Feb. 2001.
LELAI ZHOU (Member, IEEE) received the B.Eng. degree in mechanical engineering and automation from the University of Science and Technology Beijing, China, in 2006, the M.Eng. degree in mechanical design and theory from Beihang University, China, in 2009, and the Ph.D. degree in mechanical engineering from Aalborg University, Denmark, in 2012. He is currently an Associate Professor with the School of Control Science and Engineering, Shandong University. His main research interests include legged robot, robotic arm, and human-centered design.

YIBIN LI (Member, IEEE) received the B.S. degree in automation from Tianjin University, China, in 1982, the M.S. degree in electrical automation from the Shandong University of Science and Technology, China, in 1990, and the Ph.D. degree in automation from Tianjin University, in 2008. From 1982 to 2003, he worked at the Shandong University of Science and Technology, China. Since 2003, he has been the Director of the Center for Robotics, Shandong University. His research interests include bionic robots, control of intelligent robots, machine learning, and intelligence architecture.

XUEWEN RONG (Member, IEEE) received the bachelor’s and master’s degrees from the Shandong University of Science and Technology, China, in 1996 and 1999, respectively, and the Ph.D. degree from Shandong University, China, in 2013. He is currently a Professor with the School of Control Science and Engineering, Shandong University. His research interests include bionic robotics, mechatronics, hydraulic servo transmission, and control systems.

* * *