Real-Time Constrained Nonlinear Model Predictive Control for Dynamic Legged Locomotion

Joon-Ha Kim, Seungwoo Hong, Hae-Won Park

***Abstract*—In this paper, we present a novel constrained nonlinear model predictive control framework for legged locomotion. We formulate the orientation error as in the manifold, and derive all necessary derivatives for the gradient and Gauss-Newton Hessian approximation. Furthermore, we extend our previous work to calculate Gauss-Newton Hessian approximation of orientation error in the objective function efficiently. Thanks to the fast nonlinear programming solver introduced in our previous work, we can implement various dynamic legged locomotion in real-time manner. An efficient algorithm to calculate Gauss-Newton Hessian approximation is newly proposed which reduces overall computational time of Gauss-Newton algorithm significantly. We tested the proposed method on a number of benchmark dynamic motion planning problems including nonlinear model predictive control of quadrupedal robots with twelve states and twelve control inputs. The combined framework offers reliable and robust performance on all the tested problems, demonstrating its capability to control a wide range of nonlinear dynamic systems.**

I. Introduction

Animals in real world are capable of traversing in a precarious environment. For instance, an ibex can scale up nearly vertical cliffs by controlling and coordinating their dexterous hooves. Those incredible phenomena in nature motivates the realm of legged robotics to mimick the highly dynamic locomotion that animals can achieve. However, control sophisticated dynamic locomotion for legged robots is a demanding problem due to the nonlinear dynamics and the underactuation of the floating base that can only be controlled indirectly by the internal motion of the robot and the external wrenches exerted on the robot. This difficulty is further complicated by the constraints such as friction cone constraints that should be imposed on those external wrenches to avoid slip motion occured. One of the promising approaches that can solve this kind of problem is an optimization based approach that has shown remarkable performance recently.

The application of MPC on legged robots can be classified into two large groups, convex and nonconvex MPC approaches. Applications of convex MPC from humanoids [1], [2], [3],

In general, those two approaches impose a tradeoff between the model accuracy and computational efficiency. On one hand, convex MPC approaches enable a fast calculation, but their accuracy is deteriorated by approximation of nonlinear dynamics. On the other hand, nonconvex MPC approaches, based on nonlinear optimization, are accurate, but computationally demanding.

Since computational efficiency is important for controlling dynamic legged locomotion, many researchers choose to approximate the nonlinear dynamics as affine one. Since the convex MPC can be implemented in a real-time manner.

Linear inverted pendulum model (LIPM), centroidal dynamics, single rigid body dynamics.

Single rigid body dynamics offers a compromise between accuracy and efficiency; however, it is not clear how to parameterize the orientation to perform dynamic maneuvers such a back-flip and wall-climbing motions.

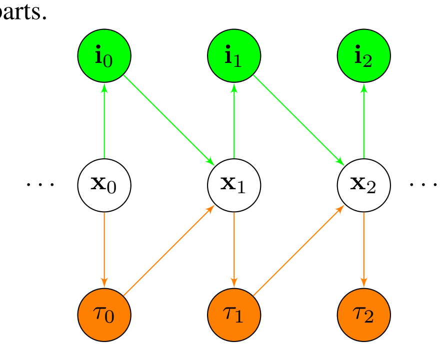
The above researches adopted Euler angles as the global parameterization for rotations. Using Euler angles as parameterization for rotations is not properly invariant under the action of rigid transformations [4]. Moreover, Euler angles are known to have singularities. In this study, we address the manifold structure of the rotation group $\mathrm{SO}(3)$.

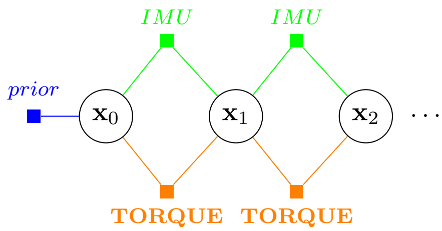
In this study, we show that it is possible to overcome this tradeoff. First, we parameterize the orientation error in tangent space of manifold. Second, we propose a novel MPC framework that enables real-time calculation of optimal solutions by using a novel NLP solvers.

The first step toward this goal is the development of all necessary Jacobians for the optimization. Furthermore, we are able to derive all necessary Jacobians in analytic form: specifically, we report the analytic Jacobians of the sigle rigid body dynamics. Compared with local parameterization or direct linearization on manifold, our method is ...

II. State and Measurement Definition

In this section, we define the state and measurement model of legged robot state estimation. Consider an optimal control problem of discrete-time deterministic system that consists of states $\mathbf{x}_k \in \mathbb{R}^n$ and control inputs $\mathbf{u}_k \in \mathbb{R}^m$ with a finite time horizon N. This optimal control problem consists of three parts.





III. ASDFASDFADSF

In this section, we define the state and measurement model of legged robot state estimation. Consider an optimal control problem of discrete-time deterministic system that consists of states $\mathbf{x}_k \in \mathbb{R}^n$ and control inputs $\mathbf{u}_k \in \mathbb{R}^m$ with a finite time horizon N. This optimal control problem consists of three parts. The first part is a nonlinear model that describes the dynamics of the system given as

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \tag{1}$$

where $k \in \{0, 1, \cdots, N\}$, and $\mathbf{x}_0$ is a given initial state. The second part is the objective function J, which is the sum of weighted deviations of the states and control inputs from a set of desired quantities in a least-squares sense over the entire time horizon considered

$$J(\mathbf{x}, \mathbf{u}) = \sum_{k=1}^{N} \frac{1}{2} \|\mathbf{r}_{\mathbf{x}_k}\|^2_{\widetilde{\mathbf{Q}}_k} + \sum_{k=0}^{N-1} \frac{1}{2} \|\mathbf{r}_{\mathbf{u}_k}\|^2_{\widetilde{\mathbf{R}}_k} \tag{2}$$

where $\mathbf{r}_{\mathbf{x}_k} = \mathbf{h}(\mathbf{x}_k)$ is a nonlinear residual error of state, and $\mathbf{r}_{\mathbf{u}_k} = \mathbf{u}_k - \mathbf{u}_k^d$ is a linear residual error of control input at time step k, respectively; $\widetilde{\mathbf{Q}}_k \in \mathbb{S}^n_{++}$ and $\widetilde{\mathbf{R}}_k \in \mathbb{S}^m_{++}$ are the corresponding weight matrices.

The final part contains constraints on control inputs that can be formulated as affine functions

$$\mathbf{A}_k \mathbf{u}_k \leq \mathbf{b}_k \tag{3}$$
$$\mathbf{C}_k \mathbf{u}_k = \mathbf{d}_k \tag{4}$$

Moreover, the decision variables can be parameterized with only control inputs $\mathbf{u}$ by representing the state variables $\mathbf{x}$ as a function of $\mathbf{u}$ using model dynamics (1). For notational convenience, we define $\mathbf{f}_{\mathbf{u}_k}(\mathbf{x}) := \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$ and the collection of control inputs and desired control inputs up to the time step $k-1$, $\mathbf{U}_k := [\mathbf{u}_0^T, \cdots, \mathbf{u}_{k-1}^T]^T$ and $\mathbf{U}_k^d := [\mathbf{u}_0^{d\,T}, \cdots, \mathbf{u}_{k-1}^{d\,T}]^T$. Then, the state $\mathbf{x}_k$ can be expressed as

$$\mathbf{x}_k = \boldsymbol{\phi}_k(\mathbf{U}_k) = \mathbf{f}_{\mathbf{u}_{k-1}} \circ \mathbf{f}_{\mathbf{u}_{k-2}} \circ \cdots \circ \mathbf{f}_{\mathbf{u}_0}(\mathbf{x}_0) \tag{5}$$

Substituting (5) into the objective function (2) with $\mathbf{h}_k(\mathbf{x}) := \mathbf{h}(\mathbf{x}_k)$ yields

$$J(\mathbf{U}_N) = \sum_{k=1}^{N} \frac{1}{2} \|\mathbf{h}_k(\boldsymbol{\phi}_k(\mathbf{U}_k))\|^2_{\widetilde{\mathbf{Q}}_k} + \sum_{k=0}^{N-1} \frac{1}{2} \|\mathbf{u}_k - \mathbf{u}_k^d\|^2_{\widetilde{\mathbf{R}}_k} \tag{6}$$

With the objective function (6), the finite-horizon discrete-time optimal control problem becomes

$$\begin{aligned} \min_{\mathbf{U}_N} \quad & \sum_{k=1}^{N} \frac{1}{2} \|\mathbf{h}_k(\boldsymbol{\phi}_k(\mathbf{U}_k))\|^2_{\widetilde{\mathbf{Q}}_k} + \sum_{k=0}^{N-1} \frac{1}{2} \|\mathbf{u}_k - \mathbf{u}_k^d\|^2_{\widetilde{\mathbf{R}}_k} \\ \text{s.t.} \quad & \mathbf{A}_k \mathbf{u}_k \leq \mathbf{b}_k, \quad k \in \{0, \cdots, N-1\}, \\ & \mathbf{C}_k \mathbf{u}_k = \mathbf{d}_k, \quad k \in \{0, \cdots, N-1\} \end{aligned} \tag{7}$$

which is in the form of constrained nonlinear least-squares problem.

Application of the proximal Gauss-Newton algorithm to this problem requires calculating the gradient and Gauss-Newton Hessian approximation of the objective function. The gradient of the objective function (6) is given by

$$\mathbf{J} = \sum_{k=1}^{N} \frac{\partial \boldsymbol{\phi}_k}{\partial \mathbf{U}_N}^T \widetilde{\mathbf{Q}}_k' \mathbf{h}_k(\boldsymbol{\phi}_k(\mathbf{U}_k)) + \boldsymbol{\Phi}_{\widetilde{\mathbf{R}}}(\mathbf{U}_N - \mathbf{U}_N^d) \tag{8}$$

where $\widetilde{\mathbf{Q}}_k' = (\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}})^T \widetilde{\mathbf{Q}}_k$, and $\boldsymbol{\Phi}_{\widetilde{\mathbf{R}}}$ is a block diagonal matrix formed with diagonal elements as $\{\widetilde{\mathbf{R}}_0, \cdots, \widetilde{\mathbf{R}}_{N-1}\}$. Similarly, the Gauss-Newton Hessian approximation of the objective function (6) is given by

$$\mathbf{H}_{GN} = \sum_{k=1}^{N} \frac{\partial \boldsymbol{\phi}_k}{\partial \mathbf{U}_N}^T \widetilde{\mathbf{Q}}_k'' \frac{\partial \boldsymbol{\phi}_k}{\partial \mathbf{U}_N} + \boldsymbol{\Phi}_{\widetilde{\mathbf{R}}} \tag{9}$$

where $\widetilde{\mathbf{Q}}_k'' = (\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}})^T \widetilde{\mathbf{Q}}_k (\frac{\partial \mathbf{h}_k}{\partial \mathbf{x}})$.

The first term of $\mathbf{J}$ and $\mathbf{H}_{GN}$ has $\frac{\partial \boldsymbol{\phi}_i}{\partial \mathbf{U}_N} = \begin{bmatrix} \frac{\partial \boldsymbol{\phi}_i}{\partial \mathbf{u}_0} & \cdots & \frac{\partial \boldsymbol{\phi}_i}{\partial \mathbf{u}_{N-1}} \end{bmatrix} \in \mathbb{R}^{n \times Nm}$ and $\frac{\partial \boldsymbol{\phi}_i}{\partial \mathbf{u}_j}$ is given by

$$\frac{\partial \boldsymbol{\phi}_i}{\partial \mathbf{u}_j} = \boldsymbol{\pi}_{i,j} \frac{\partial \mathbf{f}_j}{\partial \mathbf{u}} \tag{10}$$

where,

$$\boldsymbol{\pi}_{i,j} = \begin{cases} \prod_{k=j}^{i-2} \frac{\partial \mathbf{f}_{-k+j+i-1}}{\partial \mathbf{x}} & i \geq j+2 \\ \mathbf{I}^{n \times n} & i = j+1 \\ \mathbf{0}^{n \times n} & \text{otherwise} \end{cases} \tag{11}$$

with $\frac{\partial \mathbf{f}_j}{\partial \mathbf{x}} := \frac{\partial \mathbf{f}(\mathbf{x}_j, \mathbf{u}_j)}{\partial \mathbf{x}}$, $\frac{\partial \mathbf{f}_j}{\partial \mathbf{u}} := \frac{\partial \mathbf{f}(\mathbf{x}_j, \mathbf{u}_j)}{\partial \mathbf{u}}$.

The details of formulating the legged locomotion problem as a NMPC framework will be provided in III-A, III-B, III-C, III-D, and all the Jacobians necessary to calculate the gradient and Gauss-Newton Hessian approximation of the objective function will be introduced in III-E.

A. Model Dynamics

For a physically realizable motion, the dynamics of the robot should comply with the Newton's and Euler's equation of motion; namely, the rate of linear momentum and angular momentum of the robot should equal to all the external wrenches applied to the robot. By approximating a legged robot as a floating base sigle rigid body with point foot, the dynamics together with kinematics are as follows

$$\dot{\mathbf{p}} = \mathbf{v} \tag{12a}$$

$$\dot{\mathbf{v}} = \frac{1}{m}\sum_i \mathbf{f}_i + \mathbf{g} \tag{12b}$$

$$\dot{\mathbf{R}} = \mathbf{R}\cdot\widehat{\mathbf{w}} \tag{12c}$$

$$\dot{\mathbf{w}} = \mathbf{I}^{-1}\left(\mathbf{R}^T(\sum_i \mathbf{r}_i \times \mathbf{f}_i) - \mathbf{w}\times(\mathbf{I}\mathbf{w})\right) \tag{12d}$$

where $\mathbf{p} \in \mathbb{R}^3$ and $\mathbf{v} \in \mathbb{R}^3$ are the position and velocity of body COM, respectively; m is the mass of the robot; $\mathbf{f}_i \in \mathbb{R}^3$ is the ground reaction force (GRF) exerted on the i^{th} contact point; $\mathbf{g} \in \mathbb{R}^3$ is the gravitational acceleration; $\mathbf{R} \in \mathrm{SO}(3)$ is a 3-D rotation matrix, which is an element of Lie group, representing spatial orientation of the body frame $\mathcal{B}$; $\mathbf{I} \in \mathbb{R}^{3\times3}$ and $\mathbf{w} \in \mathbb{R}^3$ are the intertia tensor at the nominal posture and body angular velocity expressed in the body frame $\mathcal{B}$ respectively; $\mathbf{r}_i \in \mathbb{R}^3$ is the vector from the COM to the i^{th} contact point. Note that $\mathbf{p}$, $\mathbf{f}_i$, $\mathbf{g}$, and $\mathbf{r}_i$ are expressed in the inertial frame $\mathcal{I}$. The *hat* operator $\hat{(\cdot)} : \mathbb{R}^3 \to \mathfrak{so}(3)$ converts elements of 3-D vector into elements of Lie algebra consists of skew-symmetric matrices such that $\widehat{\mathbf{a}}\mathbf{b} = \mathbf{a}\times\mathbf{b}$ for all $\mathbf{a},\mathbf{b} \in \mathbb{R}^3$, where $\times$ is the vector cross product.

We discretize the continuous-time dynamics (12) using forward Euler integration with sampling time Δt to describe the discrete-time model dynamics (1) as follows

$$\mathbf{R}_{k+1} = \mathbf{R}_k \exp(\widehat{\mathbf{w}}_k \Delta t) \tag{13a}$$

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \dot{\mathbf{w}}_k \Delta t \tag{13b}$$

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \mathbf{v}_k\Delta t + \frac{1}{2}\dot{\mathbf{v}}_k \Delta t^2 \tag{13c}$$

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \dot{\mathbf{v}}_k \Delta t \tag{13d}$$

where $\exp : \mathfrak{so}(3) \to \mathrm{SO}(3)$ is the exponential map around the identity [5], [6], which coincides with Rodrigues' formula, converts elements of Lie algebra into elements of Lie group.

The discrete-time state of the system at time step k is now defined as

$$\mathbf{x}_k := [\mathbf{R}_k, \mathbf{w}_k, \mathbf{p}_k, \mathbf{v}_k] \in \mathcal{X} \tag{14}$$

where we use a 3-D rotation matrix, which provides global parameterization of $\mathrm{SO}(3)$, to represent the orientation of the robot; and $\mathcal{X} = \mathrm{SO}(3)\times\mathbb{R}^3\times\mathbb{R}^3\times\mathbb{R}^3$. The discrete-time control input to the system at time step k is defined in terms of GRFs

$$\mathbf{u}_k := \left[\mathbf{f}_{1_k}^T, \cdots, \mathbf{f}_{c_k}^T\right]^T \in \mathbb{R}^{3c} \tag{15}$$

where $c \in \mathbb{Z}^+$ indicates the number of legs of the robot.

B. Objective Function

As the state (14) contains a 3-D rotation matrix in which lives in a manifold $\mathrm{SO}(3)$, it is difficult to properly define the residual error in a least-squares sense. To tackle this problem, we express the orientation error in terms of the exponential coordinates, which coincides with the tangent space around the identity element of the manifold $\mathrm{SO}(3)$. Thus, we define the residual errors in (2) as $\mathbf{h}(\mathbf{x}_k) := [\mathbf{h}_{\boldsymbol{\varphi}_k}^T, \mathbf{h}_{\mathbf{w}_k}^T, \mathbf{h}_{\mathbf{p}_k}^T, \mathbf{h}_{\mathbf{v}_k}^T]^T \in \mathbb{R}^{12}$ with each term defined as

$$\mathbf{h}_{\boldsymbol{\varphi}_k} = \log(\mathbf{R}_k^{d^T}\mathbf{R}_k)^\vee \tag{16a}$$

$$\mathbf{h}_{\mathbf{w}_k} = \mathbf{w}_k - \mathbf{R}_k^T\mathbf{w}_k^d \tag{16b}$$

$$\mathbf{h}_{\mathbf{p}_k} = \mathbf{p}_k - \mathbf{p}_k^d \tag{16c}$$

$$\mathbf{h}_{\mathbf{v}_k} = \mathbf{v}_k - \mathbf{v}_k^d \tag{16d}$$

where $\mathbf{R}_k^d \in \mathrm{SO}(3)$, and $\mathbf{w}_k^d \in \mathbb{R}^3$, $\mathbf{p}_k^d \in \mathbb{R}^3$, $\mathbf{v}_k^d \in \mathbb{R}^3$ are the corresponding desired quantities represented in the inertial frame $\mathcal{I}$. The operator $\log : \mathrm{SO}(3) \to \mathfrak{so}(3)$ is the logarithm map around the identity [5], [6], which is the inverse of exponential map. The *vee* operator $(\cdot)^\vee : \mathfrak{so}(3) \to \mathbb{R}^3$ is the inverse of *hat* operator.

C. Reference Generation

In this section, the desired foot placement $\mathbf{r}_i$ required in swing leg controller as well as future reference generation will be introduced. As stated in III-A, $\mathbf{r}_i \in \mathbb{R}^3$ is the vector from the body COM to the i^{th} contact point. We use a foot placement strategy for the swing foot similar to the one introduced in [**?** 7], which combines zero net acceleration of the feedforward term [**?**] and velocity based feedback term [**?**], given as

$$\begin{aligned}\mathbf{p}_{f_i} &= \mathbf{p}_{f_i}^{ff} + \mathbf{p}_{f_i}^{fb} \\ &= \mathbf{p}_{h_i} + \frac{1}{2}\mathbf{v}^d T_{st} + \sqrt{\frac{p_z}{g}}(\mathbf{v} - \mathbf{v}^d)\end{aligned} \tag{17}$$

where $\mathbf{p}_{f_i}$ and $\mathbf{p}_{h_i}$ are the touch down location of i^{th} swing foot and the position of i^{th} hip expressed in the inertial frame $\mathcal{I}$, respectively; p_z is the height of the body COM.

The future reference for the i^{th} contact foot relative to the position of body COM at time step k is determined as $\forall i \in \{StancePhase\}, k \in \{1, \cdots, N\}$

$$\mathbf{r}_{i,k}^d = \mathbf{p}_{f_i,0} - \Delta T_{st}\mathbf{v}_0 \tag{18}$$

where ΔT_{st} is the stance time spent on the ground; $\mathbf{p}_{f_i,0}$ and $\mathbf{v}_0$ are the i^{th} contact foot and the velocity of body COM at the beginning of horizon, respectively. Note that all the vector variables in (18) are expressed in the inertial frame $\mathcal{I}$.

D. Constraints

To prevent the foot slip motion occured as well as to avoid the GRF generating excessive or negative force, we impose the linearized friction cone and box constraints on i^{th} contact foot, i.e., $\forall i \in \{\mathrm{Stance\,Phase}\}$

$$|f_{i_x}| \le \mu f_{i_z}, \quad |f_{i_y}| \le \mu f_{i_z}, \quad f_{min} \le f_{i_z} \le f_{max} \tag{19}$$

where μ is the friction coefficient, and the x-y-z subscript indicates the corresponding element of the vector. In order to make the robot comply with the desired gait sequence, we impose all feet forces in swing phase to zero

$$\mathbf{f}_i = \mathbf{0},\ \forall i \in \{\mathrm{Swing\,Phase}\} \tag{20}$$

E. Jacobian Derivation

In this section, we provide the Jacobians necessary to calculate the gradient and Gauss-Newton Hessian approximation of the objective function in (7).

Recall that the Gauss-Newton method iteratively finds the minimizer of quadratized objective function, which is expressed in terms of state deviation $\delta\mathbf{x}_k$ and control deviation $\delta\mathbf{u}_k$, to update the current iterate $\overline{\mathbf{x}}_k$ and $\overline{\mathbf{u}}_k$. For the state variable $\mathbf{R}_k \in \mathrm{SO}(3)$, we use the exponential map as the retraction on a manifold [?], i.e., $\mathbf{R}_k = \overline{\mathbf{R}}_k \exp(\widehat{\delta\boldsymbol{\varphi}_k})$ with $\delta\boldsymbol{\varphi}_k \in \mathbb{R}^3$. Throughout, variables with overline represent the nominal state at the associated time step. For the other state variables $\mathbf{p}_k, \mathbf{w}_k, \mathbf{v}_k$ living in the vector space $\mathbb{R}^3$, their corresponding deviations also belong to the vector space $\mathbb{R}^3$. Thus, we can define the state deviation at time step k as

$$\delta\mathbf{x}_k := [\delta\boldsymbol{\varphi}_k^T, \delta\mathbf{w}_k^T, \delta\mathbf{p}_k^T, \delta\mathbf{v}_k^T]^T \in \mathbb{R}^{12} \tag{21}$$

Similarly, the deviation of control input at time step k can be defined as

$$\delta\mathbf{u}_k := \left[\delta\mathbf{f}_{1_k}^T, \cdots, \delta\mathbf{f}_{c_k}^T\right]^T \in \mathbb{R}^{3c} \tag{22}$$

We now consider the residual error $\mathbf{h} : \mathcal{X} \to \mathbb{R}^{12}$ in (16). A standard approach to find the Jacobian of $\mathbf{h}(\mathbf{x}_k)$ is to introduce a basis for $\mathcal{X}$, find the Jacobian of the corresponding function, and substitute the result back into $\mathcal{X}$. Instead, we will directly derive the first-order approximation of $\mathbf{h}$ at $\overline{\mathbf{x}}_k$.

In the process of derivation, we will use the first-order approximation for the exponential and logarithm adopted from [4], [?],

$$\mathrm{Exp}(\boldsymbol{\psi} + \delta\boldsymbol{\psi}) \approx \mathrm{Exp}(\boldsymbol{\psi})\mathrm{Exp}(\mathbf{J}_r(\boldsymbol{\psi})\delta\boldsymbol{\psi}) \tag{23}$$

$$\mathrm{Log}\left(\mathrm{Exp}(\boldsymbol{\psi})\mathrm{Exp}(\delta\boldsymbol{\psi})\right) \approx \boldsymbol{\psi} + \mathbf{J}_r^{-1}(\boldsymbol{\psi})\delta\boldsymbol{\psi} \tag{24}$$

where $\mathbf{J}_r(\cdot)$ and $\mathbf{J}_r^{-1}(\cdot)$ are the right-Jacobians and inverse of right-Jacobians for exponential coordinates [6]; $\mathrm{Exp} : \mathbb{R}^3 \to \mathrm{SO}(3)$ and $\mathrm{Log} : \mathrm{SO}(3) \to \mathbb{R}^3$ are compact notations for $\exp(\cdot)$ and $\log(\cdot)$ used in [4], we will also use those compact notations for readability.

Let $\mathbf{x}_k \in \mathcal{X}$ be close to the nominal state $\overline{\mathbf{x}}_k$, and let $\delta\mathbf{x}_k$ assumed to be small. Similarly, let $\mathbf{u}_k \in \mathbb{R}^{3c}$ be close to $\overline{\mathbf{u}}_k$ with $\delta\mathbf{u}_k$ assumed to be small. We have

$$\mathbf{R}_k = \overline{\mathbf{R}}_k\mathrm{Exp}(\delta\boldsymbol{\varphi}_k) \tag{25a}$$

$$\mathbf{w}_k = \overline{\mathbf{w}}_k + \delta\mathbf{w}_k \tag{25b}$$

$$\mathbf{p}_k = \overline{\mathbf{p}}_k + \delta\mathbf{p}_k \tag{25c}$$

$$\mathbf{v}_k = \overline{\mathbf{v}}_k + \delta\mathbf{v}_k \tag{25d}$$

$$\mathbf{u}_k = \overline{\mathbf{u}}_k + \delta\mathbf{u}_k \tag{25e}$$

Now, with the above preliminaries, we will derive the Jacobians of $\mathbf{h}(\mathbf{x}_k)$ and $\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$ in the remaining section.

1) Jacobians of $\mathbf{h}(\mathbf{x}_k)$*:* First, Jacobians of $\mathbf{h}_{\boldsymbol{\varphi}_k}$, $\mathbf{h}_{\mathbf{w}_k}$, $\mathbf{h}_{\mathbf{p}_k}$ and $\mathbf{h}_{\mathbf{v}_k}$ are derived as below.

a) Jacobians of $\mathbf{h}_{\boldsymbol{\varphi}_k}$*:* Since $\mathbf{h}_{\boldsymbol{\varphi}_k}$ is a function of only $\mathbf{R}_k$, thereby the partial derivatives with respect to $\delta\mathbf{w}_k$, $\delta\mathbf{p}_k$, and $\delta\mathbf{v}_k$ are zero matrices. The first-order approximation of $\mathbf{h}_{\boldsymbol{\varphi}_k}$ at $\overline{\mathbf{R}}_k$ is

$$\begin{aligned}
&\mathbf{h}_{\boldsymbol{\varphi}_k}\left(\overline{\mathbf{R}}_k\mathrm{Exp}(\delta\boldsymbol{\varphi}_k)\right) \\
&\quad = \mathrm{Log}\left(\mathbf{R}_k^{d^T}\overline{\mathbf{R}}_k\mathrm{Exp}(\delta\boldsymbol{\varphi}_k)\right) \\
&\quad \overset{(24)}{\approx} \mathbf{R}_k^{d^T}\overline{\mathbf{R}}_k + \mathbf{J}_r^{-1}(\mathbf{R}_k^{d^T}\overline{\mathbf{R}}_k)\delta\boldsymbol{\phi}_k
\end{aligned} \tag{26}$$

b) Jacobians of $\mathbf{h}_{\mathbf{w}_k}$*:* Because $\mathbf{h}_{\mathbf{w}_k}$ is linear in $\delta\mathbf{w}_k$, thereby the partial derivative with respect to $\delta\mathbf{w}_k$ is an identity matrix. In addition, $\mathbf{h}_{\mathbf{w}_k}$ is independent of $\mathbf{p}_k$ and $\mathbf{v}_k$, thereby the partial derivatives with respect to $\delta\mathbf{p}_k$ and $\delta\mathbf{v}_k$ are zero matrices. The first-order approximation of $\mathbf{h}_{\mathbf{w}_k}$ at $\overline{\mathbf{R}}_k$ is

$$\begin{aligned}
&\mathbf{h}_{\mathbf{w}_k}\left(\overline{\mathbf{R}}_k\mathrm{Exp}(\delta\boldsymbol{\varphi}_k)\right) \\
&\quad = \overline{\mathbf{w}}_k - \overline{\mathbf{R}}_k\mathrm{Exp}(\delta\boldsymbol{\varphi}_k)^T\mathbf{w}_k^d \\
&\quad = \overline{\mathbf{w}}_k - \mathrm{Exp}(-\delta\boldsymbol{\varphi}_k)\overline{\mathbf{R}}_k^T\mathbf{w}_k^d \\
&\quad \overset{(a)}{\approx} \overline{\mathbf{w}}_k - (\mathbb{I} - \widehat{\delta\boldsymbol{\varphi}_k})\overline{\mathbf{R}}_k^T\mathbf{w}_k^d \\
&\quad \overset{(b)}{\approx} \overline{\mathbf{w}}_k - \overline{\mathbf{R}}_k^T\mathbf{w}_k^d + (\widehat{\overline{\mathbf{R}}_k^T\mathbf{w}_k^d})^T\delta\boldsymbol{\varphi}_k
\end{aligned} \tag{27}$$

where (a) we have used the first-order approximation of matrix exponential $\mathrm{Exp}(\delta\boldsymbol{\varphi}_k) \approx \mathbb{I} + \widehat{\delta\boldsymbol{\varphi}_k}$ with $\mathbb{I} \in \mathbb{R}^{3\times 3}$ representing an identity matrix; (b) we have used the property $\widehat{\mathbf{a}}\mathbf{b} = -\widehat{\mathbf{b}}\mathbf{a}$.

c) Jacobians of $\mathbf{h}_{\mathbf{p}_k}$ *and* $\mathbf{h}_{\mathbf{v}_k}$*:* It is straightforward that $\mathbf{h}_{\mathbf{p}_k}$ and $\mathbf{h}_{\mathbf{v}_k}$ are linear in $\delta\mathbf{p}_k$ and $\delta\mathbf{v}_k$ respectively, thereby the corresponding partial derivatives with respect to $\delta\mathbf{p}_k$ and $\delta\mathbf{v}_k$ are identity matrices, while the other partial derivatives are zero matrices.

As a result, the Jacobian of $\mathbf{h}(\mathbf{x}_k)$ is given by

$$\frac{\partial\mathbf{h}_k}{\partial\mathbf{x}} = \begin{bmatrix} \mathbf{J}_r^{-1}(\mathbf{R}_k^{d^T}\overline{\mathbf{R}}_k) & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ (\widehat{\overline{\mathbf{R}}_k^T\mathbf{w}_k^d})^T & \mathbb{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbb{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbb{I} \end{bmatrix} \in \mathbb{R}^{12\times 12} \tag{28}$$

2) Jacobians of $\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$*:* Next, Jacobians of $\mathbf{f}_{\boldsymbol{\varphi}_k}$, $\mathbf{f}_{\mathbf{w}_k}$, $\mathbf{f}_{\mathbf{p}_k}$ and $\mathbf{f}_{\mathbf{v}_k}$ are derived as below.

a) Jacobians of $\mathbf{f}_{\boldsymbol{\varphi}_k}$*:* We first note that $\mathbf{f}_{\boldsymbol{\varphi}_k}$ is independent of $\mathbf{p}_k$, $\mathbf{v}_k$, $\mathbf{u}_k$, thereby the partial derivatives with respect to $\delta\mathbf{p}_k$, $\delta\mathbf{v}_k$, $\delta\mathbf{u}_k$ are zero matrices. Let us focus on the remaning Jacobians with respect to $\delta\boldsymbol{\varphi}_k$ and $\delta\mathbf{w}_k$. Substituting (25a) at step time $k+1$ and k respectively into (13a) and rearranging the equation, we get

$$\begin{aligned}
&\mathrm{Exp}(\delta\boldsymbol{\varphi}_{k+1}) \\
&\quad \approx \overline{\mathbf{R}}_{k+1}^T\overline{\mathbf{R}}_k\mathrm{Exp}(\delta\boldsymbol{\varphi}_k)\mathrm{Exp}(\overline{\mathbf{w}}_k\Delta t) \\
&\quad \overset{(a)}{=} \overline{\mathbf{R}}_{k+1}^T\overline{\mathbf{R}}_k\mathrm{Exp}(\overline{\mathbf{w}}_k\Delta t)\mathrm{Exp}(\mathrm{Exp}^T(\overline{\mathbf{w}}_k\Delta t)\delta\boldsymbol{\varphi}_k) \\
&\quad \overset{(b)}{=} \mathrm{Exp}(\mathrm{Exp}^T(\overline{\mathbf{w}}_k\Delta t)\delta\boldsymbol{\varphi}_k)
\end{aligned} \tag{29}$$

where (a) we have used the property $\mathrm{Exp}(\boldsymbol{\varphi})\mathbf{R} = \mathbf{R}\mathrm{Exp}(\mathbf{R}^T\boldsymbol{\varphi})$; (b) is because the first two terms cancel each other by the fact $\overline{\mathbf{R}}_{k+1} = \overline{\mathbf{R}}_k\mathrm{Exp}(\overline{\mathbf{w}}_k\Delta t)$. Taking logarithm map on both sides of (29) yields

$$\delta\boldsymbol{\varphi}_{k+1} \approx \mathrm{Exp}^T(\overline{\mathbf{w}}_k\Delta t)\delta\boldsymbol{\varphi}_k \tag{30}$$

Similarly, plugging (25a) at step time $k+1$ and (25b) at step time k into (13a) and rearranging the equation, we have

$$\begin{aligned}
&\mathrm{Exp}(\delta\boldsymbol{\varphi}_{k+1})\\
&\quad= \overline{\mathbf{R}}_{k+1}^T\overline{\mathbf{R}}_k\mathrm{Exp}((\overline{\mathbf{w}}_k + \delta\mathbf{w}_k)\Delta t)\\
&\quad\overset{(23)}{\approx} \overline{\mathbf{R}}_{k+1}^T\overline{\mathbf{R}}_k\mathrm{Exp}(\overline{\mathbf{w}}_k\Delta t)\mathrm{Exp}(\mathbf{J}_r(\overline{\mathbf{w}}_k\Delta t)\Delta t\delta\mathbf{w}_k)\\
&\quad\overset{(a)}{=} \mathrm{Exp}(\mathbf{J}_r(\overline{\mathbf{w}}_k\Delta t)\Delta t\delta\mathbf{w}_k)
\end{aligned} \tag{31}$$

where (a) we have used the fact $\overline{\mathbf{R}}_{k+1} = \overline{\mathbf{R}}_k\mathrm{Exp}(\overline{\mathbf{w}}_k\Delta t)$. Taking logarithm map on both sides of (31) yields

$$\delta\boldsymbol{\varphi}_{k+1} \approx \mathbf{J}_r(\overline{\mathbf{w}}_k\Delta t)\Delta t\delta\mathbf{w}_k \tag{32}$$

b) Jacobians of $\mathbf{f}_{\mathbf{w}_k}$*:* Similar to the previous section, $\mathbf{f}_{\mathbf{w}_k}$ is independent of $\mathbf{p}_k$ and $\mathbf{v}_k$, thereby the partial derivatives with respect to $\delta\mathbf{p}_k$ and $\delta\mathbf{v}_k$ are zero matrices. We will focus on the remaning Jacobians with respect to $\delta\boldsymbol{\varphi}_k$, $\delta\mathbf{w}_k$ and $\delta\mathbf{u}_k$. Substituting (25b) at step time $k+1$ and (25a) at step time k into (13b) and rearranging the equation up to first-order terms, we get

$$\delta\mathbf{w}_{k+1} \approx -\mathbf{I}^{-1}\left[\widehat{\delta\boldsymbol{\varphi}_k}\overline{\mathbf{R}}_k^T[\mathbf{r}_k]\overline{\mathbf{u}}_k\right]\Delta t \tag{33}$$

$$\overset{(a)}{=} \mathbf{I}^{-1}\left[\widehat{\overline{\mathbf{R}}_k^T[\mathbf{r}_k]\overline{\mathbf{u}}_k}\right]\Delta t\delta\boldsymbol{\varphi}_k \tag{34}$$

where $[\mathbf{r}_k] := [\widehat{\mathbf{r}_{1_k}}, \cdots, \widehat{\mathbf{r}_{c_k}}] \in \mathbb{R}^{3\times 3c}$, and (a) we have used the property $\widehat{\mathbf{a}}\mathbf{b} = -\widehat{\mathbf{b}}\mathbf{a}$.

Similarly, plugging (25b) at step time $k+1$ and at step time k respectively into (13b) and rearranging the equation up to first-order terms, we have

$$\begin{aligned}
\delta\mathbf{w}_{k+1} &\approx \delta\mathbf{w}_k - \mathbf{I}^{-1}\left[\widehat{\overline{\mathbf{w}}}_k(\mathbf{I}\delta\mathbf{w}_k) + \widehat{\delta\mathbf{w}}_k(\mathbf{I}\overline{\mathbf{w}}_k)\right]\Delta t\\
&\overset{(a)}{=} \delta\mathbf{w}_k - \mathbf{I}^{-1}\left[\widehat{\overline{\mathbf{w}}}_k(\mathbf{I}\delta\mathbf{w}_k) - (\widehat{\mathbf{I}\overline{\mathbf{w}}_k})\delta\mathbf{w}_k\right]\Delta t\\
&\overset{(b)}{=} \left(\mathbb{I} - \mathbf{I}^{-1}\left[\widehat{\overline{\mathbf{w}}}_k\mathbf{I} - (\widehat{\mathbf{I}\overline{\mathbf{w}}_k})\right]\Delta t\right)\delta\mathbf{w}_k
\end{aligned} \tag{35}$$

where (a) and (b) we have used the property $\widehat{\mathbf{a}}\mathbf{b} = -\widehat{\mathbf{b}}\mathbf{a}$; and $\mathbb{I} \in \mathbb{R}^{3\times 3}$ is an identity matrix.

Plugging (25b) at step time $k+1$ and (25e) at step time k into (13b) and rearranging the equation up to first-order terms, we have

$$\delta\mathbf{w}_{k+1} \approx \mathbf{I}^{-1}\left(\overline{\mathbf{R}}_k^T[\mathbf{r}_k]\right)\Delta t\delta\mathbf{u}_k \tag{36}$$

c) Jacobians of $\mathbf{f}_{\mathbf{p}_k}$ *and* $\mathbf{f}_{\mathbf{v}_k}$*:* It is clear that $\mathbf{f}_{\mathbf{p}_k}$ and $\mathbf{f}_{\mathbf{v}_k}$ are linear in $\delta\mathbf{p}_k$ and $\delta\mathbf{v}_k$ respectively, thereby the corresponding partial derivatives with respect to $\delta\mathbf{p}_k$ and $\delta\mathbf{v}_k$ are identity matrices. By applying repeated application of the first-order approximation on (16c) and (16d), we have

$$\delta\mathbf{p}_{k+1} \approx \delta\mathbf{p}_k \tag{37}$$

$$\delta\mathbf{p}_{k+1} \approx \Delta t\delta\mathbf{v}_k \tag{38}$$

$$\delta\mathbf{p}_{k+1} \approx \frac{1}{2}(\frac{1}{m}[\mathbb{I}]\Delta t^2)\delta\mathbf{u}_k \tag{39}$$

$$\delta\mathbf{v}_{k+1} \approx \delta\mathbf{v}_k \tag{40}$$

$$\delta\mathbf{v}_{k+1} \approx (\frac{1}{m}[\mathbb{I}]\Delta t)\delta\mathbf{u}_k \tag{41}$$

where $[\mathbb{I}] := [\mathbb{I}, \cdots, \mathbb{I}] \in \mathbb{R}^{3\times 3c}$ with $\mathbb{I} \in \mathbb{R}^{3\times 3}$ representing an identity matrix.

In summary, the Jacobians of $\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$ are

$$\frac{\partial\mathbf{f}_k}{\partial\mathbf{x}} = \begin{bmatrix} \frac{\partial\delta\boldsymbol{\varphi}_{k+1}}{\partial\delta\boldsymbol{\varphi}_k} & \frac{\partial\delta\boldsymbol{\varphi}_{k+1}}{\partial\delta\mathbf{w}_k} & \mathbf{0} & \mathbf{0}\\ \frac{\partial\delta\mathbf{w}_{k+1}}{\partial\delta\boldsymbol{\varphi}_k} & \frac{\partial\delta\mathbf{w}_{k+1}}{\partial\delta\mathbf{w}_k} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbb{I} & \mathbb{I}\Delta t\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbb{I} \end{bmatrix} \in \mathbb{R}^{12\times 12} \tag{42}$$

with each partial derivatives specified as below,

$$\begin{aligned}
\frac{\partial\delta\boldsymbol{\varphi}_{k+1}}{\partial\delta\boldsymbol{\varphi}_k} &= \mathrm{Exp}^T(\overline{\mathbf{w}}_k\Delta t),\\
\frac{\partial\delta\boldsymbol{\varphi}_{k+1}}{\partial\delta\mathbf{w}_k} &= \mathbf{J}_r(\overline{\mathbf{w}}_k\Delta t)\Delta t,\\
\frac{\partial\delta\mathbf{w}_{k+1}}{\partial\delta\boldsymbol{\varphi}_k} &= \mathbf{I}^{-1}[\widehat{\overline{\mathbf{R}}_k^T[\mathbf{r}_k]\overline{\mathbf{u}}_k}]\Delta t,\\
\frac{\partial\delta\mathbf{w}_{k+1}}{\partial\delta\mathbf{w}_k} &= (\mathbb{I} - \mathbf{I}^{-1}[\widehat{\overline{\mathbf{w}}}_k\mathbf{I} - (\widehat{\mathbf{I}\overline{\mathbf{w}}_k})]\Delta t).
\end{aligned}$$

and

$$\frac{\partial\mathbf{f}_k}{\partial\mathbf{u}} = \begin{bmatrix} \mathbf{0}\\ \mathbf{I}^{-1}\left(\overline{\mathbf{R}}_k^T[\mathbf{r}_k]\right)\Delta t\\ \frac{1}{2}(\frac{1}{m}[\mathbb{I}]\Delta t^2)\\ \left(\frac{1}{m}[\mathbb{I}]\Delta t\right) \end{bmatrix} \in \mathbb{R}^{12\times 3c} \tag{43}$$

IV. Conclusion

We proposed a new trajectory optimization framework by combining Gauss-Newton algorithm and proximal algorithms to handle inequality constraints on controls. An efficient algorithm is also proposed for fast computation of Gauss-Newton Hessian approximation which achieved significant reduction in computational cost. Through simulation studies, we verified that the proposed algorithm has its benefit in both solve time and objective values for short horizon MPC problems. For long horizon MPC problems, the algorithm demonstrated reliable and robust performance providing the lowest objective values for most problem instances among the algorithms we tested even though the solve time is higher than iLQG.

References

[1] P.-B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *2006 6th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2006, pp. 137–142.

[2] H. Diedam, D. Dimitrov, P.-B. Wieber, K. Mombaur, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 1121–1126.

[3] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, "Online walking motion generation with automatic footstep placement," *Advanced Robotics*, vol. 24, no. 5-6, pp. 719–737, 2010.

[4] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2016.

[5] K. M. Lynch and F. C. Park, *Modern Robotics*. Cambridge University Press, 2017.

[6] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications*. Springer Science & Business Media, 2011, vol. 2.

[7] G. Bledt and S. Kim, "Implementing regularized predictive control for simultaneous real-time footstep and ground reaction force optimization," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6316–6323.