IEEE Robotics and Automation Letters (RAL) paper presented at
2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Fast Direct Optimal Control for Humanoids Based on Dynamics Representation in FPC Latent Space

Soya Shimizu[1], Ko Ayusawa[2,4], and Gentiane Venture[3,4]

***Abstract*—This study introduces a novel approach to Humanoid Robot Motion Generation using Functional Principal Component Analysis (FPCA) within the framework of Direct Optimal Control (DOC). FPCA efficiently compresses high-dimensional motion data, including ground reaction forces, into a low-dimensional space known as the FPC space. These low-dimensional elements preserve the essential characteristics of the original motions and allow for the synthesis of specific elements to create entirely new representations. By using these low-dimensional elements as optimization variables, we anticipated a significant reduction in computational time for motion generation. Experiments on the humanoid robot HRP-4J, considering various objective functions, demonstrated the method's applicability for generating whole-body motions. The Root Mean Square Error (RMSE) for angle data between the proposed method and conventional methods yielded an average RMSE of 0.007 rad, indicating precise motion generation without compromising data characteristics. Moreover, the proposed method's computation time was 35.8 times faster than the conventional approach.**

***Index Terms*—Human and Humanoid Motion Analysis and Synthesis, Whole-Body Motion Planning and Control, Optimization and Optimal Control.**

I. Introduction

HUMAN movement, with its extensive degrees of freedom and intricate kinematics, holds great potential for applications in robotics and scientific domains. Humanoid robots are explored for evaluating exoskeletons in labor-intensive fields like agriculture and caregiving [1], [2]. Additionally, research investigates using humanoid robots in hazardous or manual labor-intensive tasks [3], [4].

Controlling humanoid robots is challenging due to the high degree of freedom (DOF) and a floating base. The high DOF leads to complex computational procedures for motion control, while managing the floating base relies on indirect control through ground reaction forces, posing a significant obstacle in maintaining stability and behavior.

Manuscript received: October, 30, 2023; Revised January 17, 2024; Accepted February, 9, 2024.

This paper was recommended for publication by Editor Clement Gosselin upon evaluation of the Associate Editor and Reviewers' comments. This work was partly supported by JSPS Grant-in-Aid for Scientific Research (S) Number 22H05002 and (A) Number 22H00545.

[1]S. Shimizu is with Department of Mechanical Systems Engineering, Graduate School of Engineering, Tokyo University of Agriculture and Technology, Japan.

[2]K. Ayusawa is with the Human Augmentation Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Japan. k.ayusawa@aist.go.jp

[3]G. Venture is with Department of Mechanical Engineering, The University of Tokyo, Japan. venture@g.ecc.u-tokyo.ac.jp

[4]K. Ayusawa and G. Venture are with CNRS-AIST JRL (Joint Robotics Laboratory), UMI3218/RL, Japan.

Digital Object Identifier (DOI): see top of this page.

Various methodologies have been developed to simplify the control and generation of humanoid robot motion. Choreonoid [5], [6], a GUI control tool with a 3D simulator, enables precise regulation of robot actions within a virtual environment. *Motion retargeting* involves digitizing human movements through motion capture and modifying them for humanoid robots [7]–[9]. However, both approaches are labor-intensive and time-consuming, especially when generating a substantial volume of motion data.

Direct Optimal Control (DOC) offers a cost-effective alternative for motion generation, treating it as an optimization problem [10]–[12]. By adjusting coefficients in the computational program, diverse sets of motion data can be acquired, reducing the workload on motion capture. Nevertheless, DOC requires considerable computational time, making it less practical for high-degree-of-freedom motion trajectories, such as those for humanoid robots.

To overcome these challenges, we introduced a data synthesis methodology using Functional Principal Component Analysis (FPCA) [13], [14]. FPCA is a technique that condenses data into a lower-dimensional latent space, termed as FPC space, for analytical purposes. Within this FPC space, the intrinsic features of motion are preserved, enabling the assessment of motion similarity based on data distributions within that space. Additionally, the consolidation of various distinct datasets facilitates data synthesis.

Regarding data reduction, methods like autoencoders [15] and reinforcement learning [16], [17] can compress and restore nonlinear data without loss. However, both require learning, posing a bottleneck in terms of computational time. In contrast, thanks to its computational efficiency of linear dimensional reduction, FPCA swiftly reduces data dimensionality.

Previous research focused on synthesizing arm-reaching motions, emphasizing similarity and data synthesis. Assessing congruence between existing datasets and the target data allowed for linear synthesis, swiftly generating new data. This approach demonstrated operational efficiency nearly ten times faster than conventional DOC approaches [18].

Addressing non-linear movements, like whole-body actions considering ground reaction forces, poses a unique challenge. Ground reaction forces are crucial for torque identification at each joint [19], [20], playing a vital role in considering the dynamics of the robot's entire body [21], [22]. The inherent non-linearities of whole-body dynamics, along with additional complexities from rotational postures and ground contact conditions, make linear synthesis of such motions potentially destabilizing.

In this study, we introduce a method that directly incorpo-

rates DOC into the FPC space, aiming to synthesize non-linear motions and consider constraints related to ground reaction forces. The previous method [18] synthesized motions using a linear combination of latent variables, succeeding with upper-body movements but facing challenges in applying to lower-body motions due to dynamic balance constraints. Notably, those constraints consist of not only joint movements but reaction forces exerted on the robot from the environment. The simplistic synthesis approach in the latent space lacks assurance of meeting feasible constraints, especially involving reaction forces, while conventional DOC is a popular approach but time-consuming for constrained motion generation. This study presents a computationally efficient DOC approach derived from previous research [18], considering physical constraints, including reaction forces, for motion synthesis in the latent space.

The proposed methodology of data compression using FPCA is introduced in the following section. Additionally, the technique for optimal calculation within the FPC space is described in Section III. The results of the actual humanoid robot experiments are presented in Section IV.

II. Optimization problem in latent space using FPCA

This section describes the process of motion generation for the humanoid robot, commencing with an introduction to the DOC methodology, which incorporates ground reaction forces. This is followed by a discussion on the optimal problem resolved in the latent space using the FPCA technique.

A. Direct Optimal Control

In the current study, motion data $\boldsymbol{Q}$ is generated using DOC, conforming to predefined cost functions and constraints. The motion data $\boldsymbol{Q} \in \mathbb{R}^{N_T \times (N_J + N_{Jf})}$ (N_T denotes the number of time steps, N_J indicates the number of DOF of the robot, and N_{Jf} represents the dimension of ground reaction forces) incorporates four categories of variables: joint angle $\boldsymbol{q}$, angular velocity $\dot{\boldsymbol{q}}$, angular acceleration $\ddot{\boldsymbol{q}}$, and ground reaction force $\boldsymbol{f}$. Each variable of $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, $\ddot{\boldsymbol{q}} \in \mathbb{R}^{N_T \times N_J}$, and $\boldsymbol{f} \in \mathbb{R}^{N_T \times N_{Jf}}$ represents a matrix composed of discrete values arranged for each time step. The constraints include those related to equality and inequality $\boldsymbol{h}_{eq}(\boldsymbol{Q})$, $\boldsymbol{h}_{ineq}(\boldsymbol{Q})$. The given cost weight is described c_i ($0 \leqq c_i \leqq 1, \sum_{i=1}^{n} c_i = 1$, n indicates the number of cost functions), which are defined as $J_i(\boldsymbol{Q})$. Using those variables, the equation of DOC can be expressed as follows:

$$\begin{aligned} \underset{\boldsymbol{Q}}{\text{minimize}} \quad & J(\boldsymbol{Q}) = \sum_{i=1}^{n} c_i J_i(\boldsymbol{Q}) \\ \text{subject to} \quad & \boldsymbol{h}_{eq}(\boldsymbol{Q}) = \boldsymbol{0}, \quad \boldsymbol{h}_{ineq}(\boldsymbol{Q}) \leqq \boldsymbol{0} \end{aligned} \tag{1}$$

The solution of Eq.(1) is the optimized motion $\boldsymbol{Q}_{opt} = [\boldsymbol{q}_{opt}, \dot{\boldsymbol{q}}_{opt}, \ddot{\boldsymbol{q}}_{opt}, \boldsymbol{f}_{opt}]$ [23]. The constraints $\boldsymbol{h}_{eq}(\boldsymbol{Q})$, $\boldsymbol{h}_{ineq}(\boldsymbol{Q})$ comprise equations of $\boldsymbol{q}_{const}, \dot{\boldsymbol{q}}_{const}, \ddot{\boldsymbol{q}}_{const}, \boldsymbol{f}_{const}$, which denote the conditions of $\boldsymbol{q}_t, \dot{\boldsymbol{q}}_t, \ddot{\boldsymbol{q}}_t, \boldsymbol{f}_t$ at a specific time t_{const}. If the cost weight c_i is altered, various motion data $\boldsymbol{Q}_{opt}$ can be obtained.

B. Variables and B-Spline function

In this study, we model the joint angles and ground reaction forces trajectory using a combination of B-spline functional bases, enhancing robustness against noise and reducing dimensionality. The functionally represented data is utilized in FPCA in Section II-C. The four variables $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, $\ddot{\boldsymbol{q}}$, and $\boldsymbol{f}$ are represented as follows:

$$\begin{aligned} \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} = \sum_{t=1}^{N_T} \sum_{i=1}^{N_B} \boldsymbol{w}_{qi} \begin{bmatrix} b_{i,t} \\ \dot{b}_{i,t} \\ \ddot{b}_{i,t} \end{bmatrix}, \quad \boldsymbol{f} = \sum_{t=1}^{N_T} \sum_{i=1}^{N_{Bf}} \boldsymbol{w}_{fi} f_{i,t} \\ \Leftrightarrow \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}_q \\ \dot{\boldsymbol{B}}_q \\ \ddot{\boldsymbol{B}}_q \end{bmatrix} \boldsymbol{w}_q, \quad \boldsymbol{f} = \boldsymbol{B}_f \boldsymbol{w}_f \end{aligned} \tag{2}$$

where $\boldsymbol{w}_{qi} \in \mathbb{R}^{N_J}$ is expressed as a weight vector, $b_{i,t}, \dot{b}_{i,t}$, and $\ddot{b}_{i,t} \in \mathbb{R}$ represents the B-spline basis functions, and N_B denotes the number of the B-spline basis functions. $\boldsymbol{w}_{fi} \in \mathbb{R}^{N_{Jf}}$ is expressed as a coefficient matrix for $\boldsymbol{f}$, $f_{i,t} \in \mathbb{R}$ represent the B-spline basis functions for $\boldsymbol{f}$, and N_{Bf} indicates the number of the B-spline basis functions for $\boldsymbol{f}$. $\boldsymbol{B}_q$, $\dot{\boldsymbol{B}}_q$, and $\ddot{\boldsymbol{B}}_q \in \mathbb{R}^{N_T \times N_B}$ represent the matrices comprising the B-spline basis functions $b_{i,t}, \dot{b}_{i,t}$, and $\ddot{b}_{i,t}$. $\dot{\boldsymbol{B}}_q$ / $\ddot{\boldsymbol{B}}_q$ represents the first / second-order derivative of $\boldsymbol{B}_q$, and $\boldsymbol{w}_q \in \mathbb{R}^{N_B \times N_J}$ represents the weight matrix of the spline functions. $\boldsymbol{B}_f \in \mathbb{R}^{N_T \times N_{Bf}}$ represents the matrix comprising the B-spline basis functions $f_{i,t}$. $\boldsymbol{w}_f \in \mathbb{R}^{N_{Bf} \times N_{Jf}}$ represents the weight matrix of the spline functions for $\boldsymbol{f}$. According to Eq.(2), trajectory $\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}$, and $\boldsymbol{f}$ can be parameterized using coefficient parameters $\boldsymbol{w}_q$ and $\boldsymbol{w}_f$. In particular, three variables $\boldsymbol{q} \sim \ddot{\boldsymbol{q}}$ can be consolidated into one variable, $\boldsymbol{w}_q$, which is highly advantageous in terms of data reduction. Consequently, the motion data $\boldsymbol{Q}$ can be described using the B-spline matrix $\boldsymbol{B} = \left[\boldsymbol{B}_q, \dot{\boldsymbol{B}}_q, \ddot{\boldsymbol{B}}_q, \boldsymbol{B}_f\right]^T$, $\boldsymbol{w}_q$ and $\boldsymbol{w}_f$ as follows:

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \\ \ddot{\boldsymbol{q}} \\ \boldsymbol{f} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}_q \boldsymbol{w}_q \\ \dot{\boldsymbol{B}}_q \boldsymbol{w}_q \\ \ddot{\boldsymbol{B}}_q \boldsymbol{w}_q \\ \boldsymbol{B}_f \boldsymbol{w}_f \end{bmatrix} = \boldsymbol{B} \odot \boldsymbol{w} \tag{3}$$

where, $\odot$ represents the element-wise product of matrices.

The number of B-spline functions N_B and N_{Bf} can affect both the data size of the trajectory and the precision for its representation. Thus, N_B and N_{Bf} should be determined considering the balance between the data dimension and the precision of data expression.

Although the dimension of data can be reduced with functional representation, the large dimension of the optimized parameter matrix $\boldsymbol{w}_{opt} = [\boldsymbol{w}_{q,opt}, \boldsymbol{w}_{f,opt}]^T$ causes issues in analyzing the matrix. To this end, the FPCA method proposed by [13] is applied to the parameter matrix $\boldsymbol{w}_{opt}$ for data compression. In the following subsection, the conversion mechanism of motion data $\boldsymbol{w}_{opt}$ compresses the data into the low-dimensional variable through the FPCA.

C. Dataset conversion into score in low-dimensional latent space

After applying the FPCA to the motion dataset, the FPC scores are obtained. Herein, the given motion dataset created

with k motion data $\boldsymbol{Q}_i (1 \leqq i \leqq k)$ and k values of $\boldsymbol{w}_i$ are created from $\boldsymbol{Q}_i$. If FPCA is applied to the dataset $\boldsymbol{w}_{data} = [\boldsymbol{w}_1, \cdots, \boldsymbol{w}_k]$, the FPC space composed of FPC scores $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_k]$ will be constructed. In addition, the conversion matrix $\boldsymbol{M} \in \mathbb{R}^{N_{origin} \times N_{origin}} (N_{origin} = N_J \times N_B + N_{Jf} \times N_{Bf})$ is constructed from $\boldsymbol{B}_q$, $\boldsymbol{B}_f$, $\boldsymbol{w}_q$, and $\boldsymbol{w}_f$. Each FPC score $\boldsymbol{x}_i$ can be expressed as follows:

$$\boldsymbol{x}_i = \boldsymbol{M}(\boldsymbol{w}_i - \overline{\boldsymbol{w}}) \tag{4}$$

where $\overline{\boldsymbol{w}}$ refers to the mean value of the coefficients $\boldsymbol{w}_i$ depending on the given dataset. The specific calculations for FPCA, namely the computation of FPC scores, $\boldsymbol{X}$, and the conversion matrix, $\boldsymbol{M}$, and the mean bias, $\overline{\boldsymbol{w}}$, are detailed in [24]. Based on Eq.(4), $\boldsymbol{x}_i$ reflects the simple linear combination of $\boldsymbol{M}$ and $\boldsymbol{w}_i$, and thus, $\boldsymbol{x}_i$ can be converted into the B-spline coefficients $\boldsymbol{w}_i$, for which Eq.(4) can be rewritten as follows:

$$\boldsymbol{w}_i = \boldsymbol{M}_{pinv}\boldsymbol{x}_i + \overline{\boldsymbol{w}} \tag{5}$$

where $\boldsymbol{M}_{pinv}$ indicates the pseudo inverse of $\boldsymbol{M}$. Based on Eq.(3), Eq.(5), the motion data $\boldsymbol{Q}_i$ can be rewritten as follows:

$$\boldsymbol{Q}_i = \boldsymbol{B} \odot (\boldsymbol{M}_{pinv}\boldsymbol{x}_i + \overline{\boldsymbol{w}}) \tag{6}$$

In this context, the FPC scores are divided into $\boldsymbol{x}_i = [\widehat{\boldsymbol{x}}_i, \widetilde{\boldsymbol{x}}_i]^T$. $\widehat{\boldsymbol{x}}_i$ comprises the 1st principal component through the $N_{compressed}$-th principal component, while $\widetilde{\boldsymbol{x}}_i$ encompasses the remaining $(N_{compressed}+1)$-th principal component through the N_{origin}-th principal component. The size of $N_{compressed}$ is determined based on the contribution rates of each principal component obtained when calculating $\widehat{\boldsymbol{x}}_i$ [24]. Additionally, the pseudo inverse of the conversion matrix $\boldsymbol{M}_{pinv}$ is partitioned as $\boldsymbol{M}_{pinv} = [\widehat{\boldsymbol{M}}_{pinv}, \widetilde{\boldsymbol{M}}_{pinv}]$. $\widehat{\boldsymbol{M}}_{pinv} (\in \mathbb{R}^{N_{origin} \times N_{compressed}})$ is obtained by extracting 1st column to $N_{compressed}$-th column from the original matrix $\boldsymbol{M}_{pinv}$, and $\widetilde{\boldsymbol{M}}_{pinv} (\in \mathbb{R}^{N_{origin} \times (N_{origin} - N_{compressed})})$ is obtained by extracting columns $(N_{compressed} + 1)$-th column to N_{origin}-th column from the original matrix $\boldsymbol{M}_{pinv}$. The B-spline coefficients $\boldsymbol{w}_i$ can be expressed as follows:

$$\boldsymbol{w}_i = [\widehat{\boldsymbol{M}}_{pinv}, \widetilde{\boldsymbol{M}}_{pinv}] \begin{bmatrix} \widehat{\boldsymbol{x}}_i \\ \widetilde{\boldsymbol{x}}_i \end{bmatrix} + \overline{\boldsymbol{w}} \tag{7}$$

Eq.(6) can be rewritten based on Eq.(7):

$$\begin{aligned} \boldsymbol{Q}_i &= \boldsymbol{B} \odot ([\widehat{\boldsymbol{M}}_{pinv}, \widetilde{\boldsymbol{M}}_{pinv}] \begin{bmatrix} \widehat{\boldsymbol{x}}_i \\ \widetilde{\boldsymbol{x}}_i \end{bmatrix} + \overline{\boldsymbol{w}}) \\ \Leftrightarrow \boldsymbol{Q}_i &= \boldsymbol{B} \odot (\widehat{\boldsymbol{M}}_{pinv}\widehat{\boldsymbol{x}}_i + \overline{\boldsymbol{w}}) + \boldsymbol{B} \odot \widetilde{\boldsymbol{M}}_{pinv}\widetilde{\boldsymbol{x}}_i \\ &= \widehat{\boldsymbol{Q}}_i + \Delta\boldsymbol{Q}_i \end{aligned} \tag{8}$$

The first term, $\widehat{\boldsymbol{Q}}_i$, represents the approximated motion data generated from the low-dimensional score $\widehat{\boldsymbol{x}}_i$, while the second term, $\Delta\boldsymbol{Q}_i$, represents the value generated from $\widetilde{\boldsymbol{x}}_i$. By appropriately determining $N_{compressed}$ based on the contribution rates of principal components, the term, $\Delta\boldsymbol{Q}_i$, in Eq.(8) can be neglected (i.e., $\Delta\boldsymbol{Q}_i \approx \boldsymbol{0}$). Therefore, excluding the term, $\Delta\boldsymbol{Q}_i$, we obtain the following equation:

$$\boldsymbol{Q}_i \approx \widehat{\boldsymbol{Q}}_i = \boldsymbol{B} \odot (\widehat{\boldsymbol{M}}_{pinv}\widehat{\boldsymbol{x}}_i + \overline{\boldsymbol{w}}) \tag{9}$$

This formula allows obtaining the approximated motion data, $\widehat{\boldsymbol{Q}}_i$, instead of the original data, $\boldsymbol{Q}_i$, by utilizing the low-dimensional score, $\widehat{\boldsymbol{x}}_i$. Moreover, Eq.(9) implies that by providing an arbitrary $N_{compressed}$-dimensional FPC score $\widehat{\boldsymbol{x}}$, it is possible to generate new motion data $\widehat{\boldsymbol{Q}}$.

D. Optimization problem in FPC space

The motion data $\boldsymbol{Q}$ can be simply calculated based on the FPC score $\boldsymbol{x}$ using the matrices: $\boldsymbol{M}_{pinv}$, $\overline{\boldsymbol{w}}$, and $\boldsymbol{B}$. Consequently, Eq.(1) can be rewritten as the optimization problem for the FPC score $\boldsymbol{x}$:

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \widehat{J}(\boldsymbol{x}) = \sum_{i=1}^{n} c_i \widehat{J}_i(\boldsymbol{x}) \\ \text{subject to} \quad & \widehat{\boldsymbol{h}}_{eq}(\boldsymbol{x}) = \boldsymbol{0}, \quad \widehat{\boldsymbol{h}}_{ineq}(\boldsymbol{x}) \leqq \boldsymbol{0} \end{aligned} \tag{10}$$

where $\widehat{J}(\boldsymbol{x})$ and $\widehat{J}_i(\boldsymbol{x})$ represent the cost functions for the FPC score $\boldsymbol{x}$, $\widehat{\boldsymbol{h}}_{eq}(\boldsymbol{x})$ and $\widehat{\boldsymbol{h}}_{ineq}(\boldsymbol{x})$ denote the equality and inequality constraints for $\boldsymbol{x}$, respectively. Based on the Eq.(1) and Eq.(10), the initial optimization problem concerning $\boldsymbol{Q}$ can be reformulated as that concerning $\boldsymbol{x}$. When considering the compressed FPC score $\widehat{\boldsymbol{x}}$, from Eq.(9), the Eq.(10) becomes an approximated optimization problem for $\widehat{\boldsymbol{x}}$, as described below:

$$\begin{aligned} \underset{\widehat{\boldsymbol{x}}}{\text{minimize}} \quad & \widehat{J}(\widehat{\boldsymbol{x}}) = \sum_{i=1}^{n} c_i \widehat{J}_i(\widehat{\boldsymbol{x}}) \\ \text{subject to} \quad & \widehat{\boldsymbol{h}}_{eq}(\widehat{\boldsymbol{x}}) = \boldsymbol{0}, \quad \widehat{\boldsymbol{h}}_{ineq}(\widehat{\boldsymbol{x}}) \leqq \boldsymbol{0} \end{aligned} \tag{11}$$

The compressed FPC score $\widehat{\boldsymbol{x}}$ reflects the low-dimensional variable, and the calculation of Eq.(11) can be faster than that of Eq.(1), DOC with the motion data $\boldsymbol{Q}$.

As described earlier, the process is utilized to create the motion data from the given cost weights c_i and matrices ($\boldsymbol{M}_{pinv}$, $\overline{\boldsymbol{w}}$, and $\boldsymbol{B}$) instead of DOC.

III. VARIABLE SPECIFICATION

This section provides a more detailed explanation of the variables $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, $\ddot{\boldsymbol{q}}$, $\boldsymbol{f}$ and cost functions and constraints used in the optimization problem.

A. Objective functions of DOC

To comprehensively assess humanoid motion, we considered three objective functions: acceleration of the center of mass (COM), torque of robot links, and power of links. This choice aims to prevent an explosive increase in the data when using four or more types, as observed in Section IV-B. While the previous work [18] concentrated on the arm movements, this paper shifts its focus by changing the evaluation from the jerks of robot links to the acceleration of COM during the whole-body movements. COM is adopted as a substitute for the zero moment point (ZMP), used as an objective function for generating whole-body movements like walking [25]. In this study, with little change in the xy coordinates of total mass points of the robot, we deemed using COM as an objective function acceptable. Other cost functions, often used for evaluating human or humanoid movements [23], [26], can be formulated as follows:

(J_1) COM acceleration: $J_1 = \sum_i^{N_J} \sum_t^{N_T} \|\boldsymbol{g}(q_{i,t}, \dot{q}_{i,t}, \ddot{q}_{i,t})\|^2$

(J_2) Torque: $J_2 = \sum_i^{N_J} \sum_t^{N_T} \|\boldsymbol{\tau}(q_{i,t}, \dot{q}_{i,t}, \ddot{q}_{i,t}, f_{i,t})\|^2$

(J_3) Power: $J_3 = \sum_i^{N_J} \sum_t^{N_T} (\dot{q}_{i,t}\boldsymbol{\tau})^2$

where $\boldsymbol{g}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}})$ represents COM acceleration that can be computed from the joint angles and their time derivatives. The function $\boldsymbol{\tau}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}, \boldsymbol{f})$ indicates the joint torques obtained from the following inverse dynamics computation:

$$\begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{\tau}\end{bmatrix} = \begin{bmatrix}\boldsymbol{M}_b(\boldsymbol{q})\\ \boldsymbol{M}_r(\boldsymbol{q})\end{bmatrix}\ddot{\boldsymbol{q}} + \begin{bmatrix}\boldsymbol{C}_b(\boldsymbol{q}, \dot{\boldsymbol{q}})\\ \boldsymbol{C}_r(\boldsymbol{q}, \dot{\boldsymbol{q}})\end{bmatrix} + \begin{bmatrix}\boldsymbol{G}_b(\boldsymbol{q})\\ \boldsymbol{G}_r(\boldsymbol{q})\end{bmatrix} + \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{I}_r\end{bmatrix}\ddot{\boldsymbol{q}} + \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{V}_r\end{bmatrix}\dot{\boldsymbol{q}} - \begin{bmatrix}\boldsymbol{J}_b^T\\ \boldsymbol{J}_r^T\end{bmatrix}\sum_i^m \boldsymbol{f}_i \quad (12)$$

where $\boldsymbol{M}(\boldsymbol{q})$ indicates the inertia matrix, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ represents the Coriolis matrix, and $\boldsymbol{G}(\boldsymbol{q})$ reflects the gravity vector. $\boldsymbol{I}$ denotes the inertia matrix of joint motors of the humanoid robot. $\boldsymbol{V}$ includes the viscosity of the joint motors. $\boldsymbol{J}^T$ converts m descrete ground reaction forces $\boldsymbol{f}_i (i = 1, \cdots, m)$ into generalized forces. $*_b$ represents the matrix related to the floating base, whereas $*_r$ pertains to matrices associated with other robotic components. These objective functions $J_1 \sim J_3$, are substituted into Eq.(1) and Eq.(11). When substituting $J_1 \sim J_3$ into Eq.(11), variables $\boldsymbol{q} \sim \ddot{\boldsymbol{q}}$ in $J_1 \sim J_3$ are replaced with variables $\widehat{\boldsymbol{q}} \sim \ddot{\widehat{\boldsymbol{q}}}$ generated from the lower-dimensional variable $\widehat{\boldsymbol{x}}$ according to Eq.(9).

B. Constraints of DOC

To enhance the stability of the robot's motion, several physical constraint conditions are introduced. Equality constraints primarily involve boundary conditions, whereas inequality constraints predominantly concern factors such as the range of motion of the robot's joint actuators.

1 Equality Constraints:

(e1) Boundary conditions: $\begin{cases}\boldsymbol{q}_0, \boldsymbol{q}_T = \boldsymbol{q}_{init}\\ \dot{\boldsymbol{q}}_0, \dot{\boldsymbol{q}}_T, \ddot{\boldsymbol{q}}_0, \ddot{\boldsymbol{q}}_T = \boldsymbol{0}\end{cases}$

(e2) Constraints at specific times: $\begin{cases}\dot{\boldsymbol{q}}_k = \dot{\boldsymbol{q}}_{des}\\ \boldsymbol{p}_k = \boldsymbol{p}_{des}\end{cases}$

(e3) Contact constraints about feet: $\begin{cases}\boldsymbol{e}_{feet} = \boldsymbol{e}_{des}\end{cases}$

(e4) Constraint about under-actuated floating base: $\begin{cases}\boldsymbol{M}_b\ddot{\boldsymbol{q}} + \boldsymbol{C}_b + \boldsymbol{G}_b - \boldsymbol{J}_b^T \sum_i^m \boldsymbol{f}_i = \boldsymbol{0}\end{cases}$

2 Inequality Constraints:

(i1) Range of robot motion: $\begin{cases}\boldsymbol{q}_{min} \leqq \boldsymbol{q} \leqq \boldsymbol{q}_{max}\\ \boldsymbol{\tau}_{min} \leqq \boldsymbol{\tau} \leqq \boldsymbol{\tau}_{max}\\ \boldsymbol{F}_{min} \leqq \boldsymbol{F} \leqq \boldsymbol{F}_{max}\end{cases}$

(i2) COM range: $\begin{cases}y_{const} \leqq y_{com}\\ Ax_{com} - b \leqq y_{com} \leqq Ax_{com} + b\end{cases}$

(i3) Ground reaction force: $\begin{cases}\boldsymbol{A}_{face}\boldsymbol{F} \leqq 0\end{cases}$

where $\boldsymbol{p}_k$ represents the positions of arbitrary links, $\boldsymbol{e}_{feet}$ indicates the quaternion of both feet, $\boldsymbol{F}$ denotes a vector composed of translational forces $\boldsymbol{f}_0$ and moments $\boldsymbol{m}_0$ on the ground ($\boldsymbol{F} = [\boldsymbol{f}_0, \boldsymbol{m}_0]^T$), y_{const}, A, b represent the constant values with respect to the COM, and μ represents the coefficient of friction of the floor, x_{com}, y_{com} denotes the coordinates of the COM. (e2) signifies that the motion trajectory passes through the desired values ($*_{des}$) at a specific instant. (e3) represents that the position and attitude of both feet are set. (e4) represents the constraints related to the floating base. Based on Eq.(12), the torque applied to the floating base is 0 at all instances while the robot is in motion. (i2) represents the constraint conditions related to the position of the overall COM. This setting ensures that the COM does not deviate significantly from the support polygon formed by both feet. (i3) represents the inequality constraints related to the vector of the ground reaction force $\boldsymbol{F}$. For the matrix $\boldsymbol{A}_{face}$, please refer to the appendix.

IV. Experimental results of motion synthesis

In this section, we present an experimental evaluation conducted using the humanoid robot HRP-4J.

A database is first generated from different weight combinations, and FPCA is applied to obtain the matrix $\boldsymbol{M}$ for reconstructing the original motion data. Optimization challenges in the low-dimensional space are then addressed, and the solutions are transformed back into motion data for execution by the humanoid robot. Throughout these steps, metrics like computational speed and motion data fidelity to the original are evaluated.

A. Various experimental settings

This subsection outlines the specifications of the robot and delineates the detailed settings for motion data.

HRP-4J is a multi-purpose humanoid robot equipped with 34 degrees of freedom (DOF), developed by the National Institute of Advanced Industrial Science and Technology [27]. For the scope of DOC control, we consider 18 dimensions, comprising both legs (6 DOFs each) and the waist (an additional 6 dimensions as the floating base). Therefore, $N_J = 18$. Additionally, the ground reaction forces $\boldsymbol{f}$ possess 12 dimensions (6 dimensions for each leg), and $N_{Jf} = 12$. Accordingly, the number of B-spline basis functions were set to be 20 for the angular variables N_B and the ground reaction forces N_{Bf}. Consequently, the original dimension of the parameter matrix $\boldsymbol{w}$ becomes $N_{origin} = N_J \times N_B + N_{Jf} \times N_{Bf} = 600$.

The executed motion involves a lunge with forward and backward leg movements. Advancing the right leg while retracting the left, the robot lowers its waist over 3 s, pausing for 1 s at the lowest point to minimize body sway. It then elevates its waist over 3 s, completing the motion in a total duration of 7 s.

The details of the constraints introduced in III-B are presented below. The units of length and time are in meters and seconds, respectively. The horizontal axis of the robot is denoted as x, the sagittal axis as y, and the vertical axis as z. For items without specific notation, the conditions outlined in III-B are followed.

1 Equality Constraints:

TABLE I: Contribution ratio (CR) and Cumulative contribution ratio (CCR)

n-th PC	1	2	3	4	5	6	7	8	9	10
CR [%]	50.03	26.57	12.63	7.14	2.26	0.48	0.34	0.24	0.14	0.081
CCR [%]	50.03	76.60	89.23	96.37	98.63	99.11	99.45	99.69	99.83	99.91
n-th PC	11	12	13	14	15	16	17	18	19	20
CR [%]	0.069	0.0075	0.0060	0.0025	0.0014	0.00060	0.00030	0.00030	0.00030	0.00010
CCR [%]	99.98	99.9886	99.9945	99.9970	99.9984	99.9990	99.9993	99.9996	99.9999	100

TABLE II: Calculation times and RMSEs of DOC and proposed method

$\boldsymbol{c}_{sub}$	$\boldsymbol{c}_{sub1}$ [0.2, 0.6, 0.2]	$\boldsymbol{c}_{sub2}$ [0.33, 0.33, 0.33]	$\boldsymbol{c}_{sub3}$ [0.5, 0.5, 0]	$\boldsymbol{c}_{sub4}$ [0.1, 0.1, 0.8]	$\boldsymbol{c}_{sub5}$ [1, 0, 0]	mean
DOC [sec]	104210	91624	62060	42661	31370	66385
Proposed method [sec]	1403	2793	3807	1579	1091	2135
Ratio	74.3	32.8	16.3	27.0	28.8	35.8
RMSE	0.132	0.150	0.142	0.141	0.142	0.141

(e1*) Boundary conditions: $\left\{ z_{b,0} = 0 \right.$

(e2*) Constraints at specific times: $\begin{cases} \dot{\boldsymbol{q}}_{stop} & = \boldsymbol{0} \\ z_{b,stop} & = -0.04 \end{cases}$

(e3*) Contact constraints about feet:

$$\begin{cases} \boldsymbol{p}_{right} & = [0.15, 0.25, -0.6958] \\ \boldsymbol{p}_{left} & = [-0.15, -0.25, -0.6958] \\ \theta^z_{right} & = 85^\circ \\ \theta^z_{left} & = 95^\circ \end{cases}$$

2 Inequality Constraints:

(i2*) COM range:

$$\begin{cases} -0.015 \leqq y_{com} \\ \dfrac{5}{3}x_{com} - \dfrac{25}{3000} \leqq y_{com} \leqq \dfrac{5}{3}x_{com} + \dfrac{25}{3000} \end{cases}$$

where z_b denotes the z coordinate of the floating base, $*_{stop}$ represents the time interval of "stop phase" during which the robot remains stationary for 1 second upon reaching the lowest point. θ^z represents the rotation angle around the z-axis. The constraints of the quaternion of both feet $\boldsymbol{e}_{des}$ were calculated using (e3*). The initial joint angles for the lower body of HRP-4J $\boldsymbol{q}_{init}$ were calculated using inverse kinematics from the constraints (e1*) and (e3*).

B. Data compression by FPCA

The weights of cost functions $\boldsymbol{c}_{db} = [c_{db1}, c_{db2}, c_{db3}]$ were prepared in multiple instances. $c_{db1\sim db3}$ were varied in increments of 0.2 according to the following conditions, resulting in 21 combinations of $\boldsymbol{c}_{db}$.

$$\begin{cases} c_{dbi} = \{0, 0.2, 0.4, 0.6, 0.8, 1.0\} \\ c_{db1} + c_{db2} + c_{db3} = 1 \end{cases} \tag{13}$$

These generated weights $c_{db1\sim db3}$ are substituted in place of c_i in the DOC represented by Eq.(1) to create a motion dataset, and Eq.(1) is solved using MATLAB's Optimization Toolbox (fminunc) with CasADi, which is an automatic differentiation tool for gradient computations in optimization [28]. This yields the dataset $\boldsymbol{Q}_{data}$ with 21 motion data $\boldsymbol{Q}$, which is then transformed into a coefficient dataset $\boldsymbol{w}_{data}$ comprising 21 coefficient data $\boldsymbol{w}$ from Eq.(3).

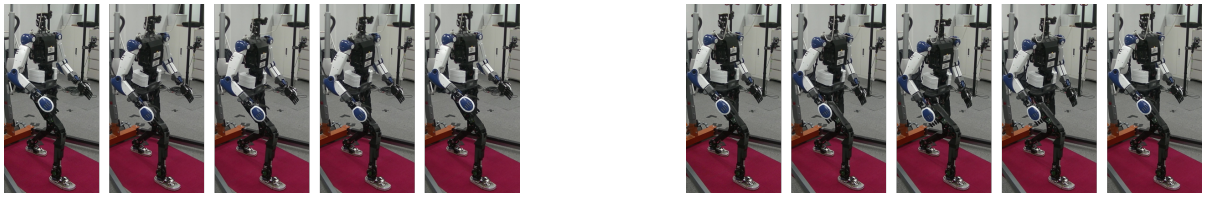
Consequently, we calculated the conversion matrix $\boldsymbol{M}$, $\overline{\boldsymbol{w}}$, FPC scores $\boldsymbol{x}$, and the contribution ratio. Table I presents the contribution ratios (CR) and cumulative contribution ratios (CCR) of each principal component (PC). Up to the 1st $\sim$ 10th PC, 99.9% of the data is explained, and up to the 1st $\sim$ 20th PC, 100% is explained. The remaining 580 dimensions ($\because N_{origin} = 600$) can be safely disregarded for data explanation.

Moreover, applying the computation of the sum of squared elements of $\Delta\boldsymbol{Q}_i = [\Delta\boldsymbol{q}_i, \Delta\dot{\boldsymbol{q}}_i, \Delta\ddot{\boldsymbol{q}}_i, \Delta\boldsymbol{f}_i] (= \boldsymbol{B} \odot \widetilde{\boldsymbol{M}}_{pinv}\widetilde{\boldsymbol{x}}_i)$ in Eq.(8) under the given conditions to a dataset consisting of 21 types of data resulted in an average angle $\Delta\boldsymbol{q}$ of 1.228×10^{-20} rad, angular velocity $\Delta\dot{\boldsymbol{q}}$ of 9.682×10^{-18} rad/s, angular acceleration $\Delta\ddot{\boldsymbol{q}}$ of 2.638×10^{-15} rad/s^2, and the ground reaction forces $\Delta\boldsymbol{f}$ of 3.542×10^{-15} N. These values are extremely small; therefore, even if we ignore $\Delta\boldsymbol{Q}_i$, it can be considered that there is minimal information loss, and the constraints can be satisfied to the same extent as the original data $\boldsymbol{Q}_i$. Based on these findings, the FPC scores compressed up to the 1st $\sim$ 20th component will be used for the subsequent optimization calculations.

C. Comparison of calculation time

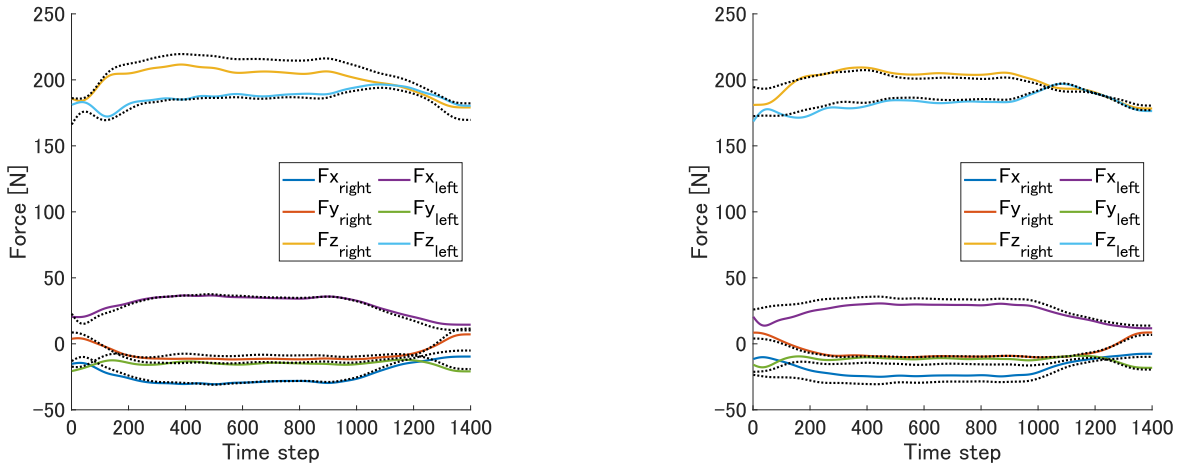
In this subsection, we compare the solving time for optimization problems in the FPC space with conventional DOC. The conventional DOC was solved as a motion optimization problem [29]. We initially chose arbitrary weight combinations $\boldsymbol{c}_{sub}$ for cost functions and computed motion data using both methods. The computational time for solving, excluding preliminary steps like dataset creation, was recorded. For this, five different $\boldsymbol{c}_{sub}$ values were prepared: $\boldsymbol{c}_{sub} = [0.2, 0.6, 0.2]$, $[0.33, 0.33, 0.33]$, $[0.5, 0.5, 0]$, $[0.1, 0.1, 0.8]$, $[1, 0, 0]$. The final $\boldsymbol{c}_{sub}$ has $c_1 = 1$, indicating minimal COM acceleration.

The results about computational time are tabulated in Table II. The table's bottom row displays the Root Mean Squared Error (RMSE) between the uncompressed data $\boldsymbol{w}$ generated by conventional DOC and the compressed data $\widehat{\boldsymbol{w}}$ produced by the proposed method. Remarkably, it becomes evident that solving low-dimensional optimization problems is, on average, 35.8 times faster than solving in the original dimensions. Moreover, the average RMSE value is 0.141, implying that the compressed data closely approximates the original uncompressed data with a high degree of accuracy. This is especially significant given that the range of $\boldsymbol{w}$ values, averaged over the five dataset, is approximately $-1.0 < \boldsymbol{w} < 1.6$.



(a) Minimal COM = [1, 0, 0] (b) $\boldsymbol{c}_{sub2} = [0.33, 0.33, 0.33]$

Fig. 1: Right leg lunge motion snapshots of HRP-4J for some combinations of weights of objective functions The robot bends to lower its upper body and then gradually returns to its initial posture.



(a) $\boldsymbol{c}_{sub1} = [0.2, 0.6, 0.2]$ (b) $\boldsymbol{c}_{sub2} = [0.33, 0.33, 0.33]$

Fig. 2: Contact forces recorded by force sensors on both feet of the robto along xyz axis of each $\boldsymbol{c}_{sub}$ motion Solid line: Compressed data (20 dimensions) Dashed line: Uncompressed data (600 dimensions)

D. Robot experiment

The data from $\boldsymbol{c}_{sub}$ in Section IV-C was applied to the HRP-4J robot. Using the robot's sensors, logs were collected for joint angles, torque, and forces applied to the foot soles. This information was then used to calculate angular velocity, angular acceleration, and power values.

1) Motion snapshots: The motion execution is shown in Figure 1, featuring two distinct motions: minimal COM acceleration (min COM) and $\boldsymbol{c}_{sub2} = [0.33, 0.33, 0.33]$. Notably, $\boldsymbol{c}_{sub2}$ was generated from the dataset. The robot consistently performs lunge movements without falling or leg slippage across all motions. Specifically, the stability of the robot during the $\boldsymbol{c}_{sub2}$ motion, generated through our proposed method, underscores its practical efficacy. The min COM motion involves a vertical downward movement of the upper body, as shown in Figure 1a. Additionally, the $\boldsymbol{c}_{sub2}$ motion resembles the min COM motion, irrespective of the $\boldsymbol{c}$ value (Figure 1b), indicating the pronounced influence of min COM during the DOC process.

2) Trajectory analysis: Subsequent to this, a comparative analysis of the logged data from the robot's sensors will be undertaken. Figure 2 portrays the outcomes of superimposing the force sensor logs for two different motions-those generated using conventional DOC (uncompressed data) and those using DOC within the FPC space (compressed data)-both originating from cost function weight combinations $\boldsymbol{c}_{sub1} = [0.2, 0.6, 0.2]$ and $\boldsymbol{c}_{sub2} = [0.33, 0.33, 0.33]$. Force sensor readings from both foot soles closely align with the uncompressed data. The RMSE values corresponding to each trajectory are summarized in Table III. Given the scale of the values, it is apparent that errors, excluding those in torque, are relatively minor, thereby confirming that the compressed data closely approximates the original uncompressed data with a high degree of accuracy. The somewhat larger errors in torque measurements can be attributed to the inability to control torque $\boldsymbol{\tau}$ directly through DOC; consequently, errors originating from approximations in $\boldsymbol{q}$ and the force $\boldsymbol{f}$ accumulate in the torque data.

TABLE III: RMSEs of each value of each $\boldsymbol{c}_{sub}$ motion

$\boldsymbol{c}_{sub}$	$\boldsymbol{c}_{sub1}$ [0.2, 0.6, 0.2]	$\boldsymbol{c}_{sub2}$ [0.33, 0.33, 0.33]	mean
Angle [rad]	0.0113	0.0041	0.0077
Velocity [rad/s]	0.0011	0.0024	0.0018
Acceleration [rad/s^2]	0.0064	0.0095	0.0080
Torque [N · m]	2.2864	2.6737	2.4800
Power [W]	0.1529	0.2910	0.2219
Force [N]	4.5744	4.1997	4.3871

3) Cost comparison: The torque and power costs of the motion data were computed using equations J_2 and J_3 in III-A. Since the COM measurements were not available in the robot logs, the cost related to COM acceleration J_1 was excluded due to computational complexity. The results are presented in bar graphs in Figure 3, showcasing outcomes for 11 different

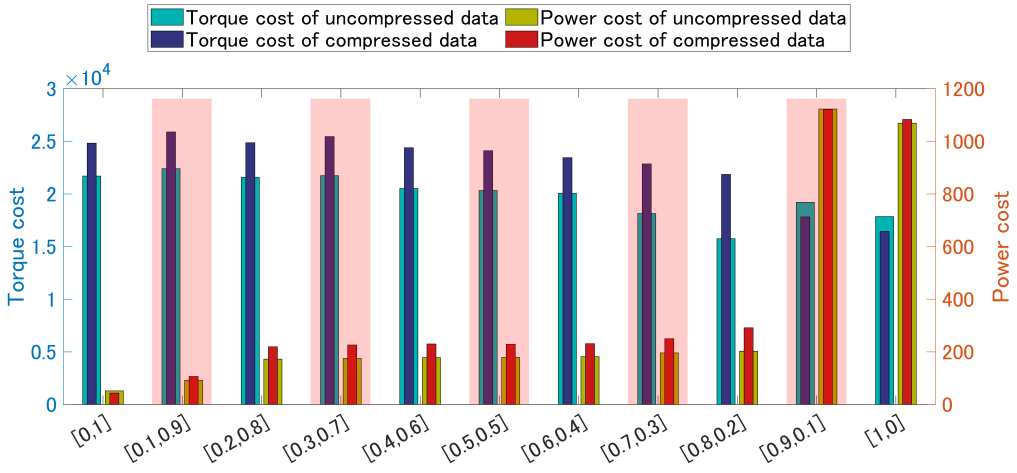


Fig. 3: Actual torque and power costs calculated from motion for each weight combination. The labels on the x-axis represent the weights of the torque and power objective functions ($\boldsymbol{c} = [c_2, c_3]$). Min power is [0, 1], and min torque is [1, 0]. The highlighted bar graphs in pink represent combinations not present in the dataset.

combinations of cost variables, including some not present in the dataset. The compressed data display changes in cost corresponding to the weight of the objectives. Notably, cost trends for combinations not in the dataset generally align with expected magnitude relationships. For instance, an increase in the torque objective function weight (c_2) results in reduced torque cost, and an increase in the power objective function weight (c_3) leads to decreased power cost. Discrepancies in costs (e.g., torque cost for [0,1] and [0.1,0.9], and power cost for [0.9,0] and [1,0]) may stem from factors such as joint motor viscosity, experimental conditions, including floor and robot motor temperatures, and log filtering strength.

V. Conclusion and Discussion

In this study, we propose a novel approach to Direct Optimal Control (DOC) for efficient whole-body motion generation in humanoid robots. Leveraging Functional Principal Component Analysis (FPCA) within a latent space termed FPC space, we achieve faster computation of DOC without compromising accuracy compared to traditional methods. FPCA captures low-dimensional data, including robot joint angles and ground reaction forces, which can be reconstructed into the original high-dimensional motion data. By formulating optimization variables within this low-dimensional space, we can generate motion data while ensuring the robot's stability constraints.

To assess the feasibility of our proposed approach, we conducted experiments with the humanoid robot HRP-4J. Building a database and determining crucial parameters for FPCA-based DOC, we executed DOC in the FPC space, achieving motion data generation with a computation time 35.8 times faster than traditional DOC methods. Operating HRP-4J using the generated data resulted in stable performance without falls, even with combinations of cost function weights not present in the initial dataset. The RMSE values between the proposed method and traditional method for angle, angular velocity, and angular acceleration were remarkably small (0.0077 rad, 0.0018 rad/s, and 0.0080 rad/s^2, respectively), indicating close approximation of motions. This suggests that motions generated using our proposed methodology closely approximate those from traditional DOC methods in their original, uncompressed dimensions.

One limitation of this study is the utilization of only three basic objective functions to generate the motion dataset. To encompass a broader spectrum of motions, it is essential to incorporate a more diverse set of objective functions. However, the combinations of weights for these functions increase exponentially, leading to a significant increase in preparation time. A notable challenge lies in identifying which objective functions are essential for generating the desired range of motions. Moreover, the current constraints applied to the motion data are assumed to remain constant throughout the motion. In experiments involving activities such as walking, the physical constraints of the motion, such as conditions related to the contact of the left and right legs, and other factors like the center of mass (COM) and zero moment point (ZMP), change constantly. When compressing with FPCA, it becomes a topic of discussion whether it is appropriate to aggregate these continuously changing conditions into a single point in the latent space. In the future, we plan to utilize the fast computational capabilities provided by the proposed methodology for real-time DOC development. Additionally, we aim to expand this approach to generate walking motions by constructing the latent space while considering the transition of foot contact conditions.

Appendix

Fundamentally, the ground reaction force is generated from numerous points on the sole of the foot. However, for program implementation and computational speed, n discrete forces $\boldsymbol{f}_i = [f_x, f_y, f_z]^T$ were applied to approximate the ground reaction forces. The inequalities for f_x, f_y, and f_z, are defined by Eq. (14), where μ denotes the friction coefficient of the ground. Eq.(15) represents the inequalities in the face format of $\boldsymbol{f}$'s convex polyhedron.

$$\begin{cases} -\mu f_z \leqq f_x \leqq \mu f_z \\ -\mu f_z \leqq f_y \leqq \mu f_z \\ 0 \leqq f_z \end{cases} \Leftrightarrow \begin{bmatrix} -1 & 0 & -\mu \\ 1 & 0 & -\mu \\ 0 & -1 & -\mu \\ 0 & 1 & -\mu \\ 0 & 0 & -1 \end{bmatrix} \boldsymbol{f}_i \leqq \mathbf{0}$$
$$\Leftrightarrow \boldsymbol{B}_i \boldsymbol{f}_i \leqq \mathbf{0} \quad (14)$$

$$\begin{bmatrix} \boldsymbol{B}_1 & & \\ & \ddots & \\ & & \boldsymbol{B}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_1 \\ \vdots \\ \boldsymbol{f}_n \end{bmatrix} \Leftrightarrow \boldsymbol{B}_{face} \boldsymbol{f} \leqq \mathbf{0} \quad (15)$$

Conforming to the convex polyhedron theory [30], $\boldsymbol{f}$ can be expressed in span format as follows:

$$\boldsymbol{f} = \boldsymbol{B}_{span} \boldsymbol{S} \geqq \mathbf{0} \quad (16)$$

where $\boldsymbol{B}_{span}$ represents the matrix in the span format of $\boldsymbol{B}_{face}$, and $\boldsymbol{S}$ indicates a collection of span vectors.

The combined force $\boldsymbol{f}_0$ and moment $\boldsymbol{m}_0$ resulting from $\boldsymbol{f}_i$ are considered to act on the specific joint. The relationship between $\boldsymbol{f}$, and $\boldsymbol{F} = [\boldsymbol{f}_0, \boldsymbol{m}_0]^T$ is expressed by:

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{E} \cdots \boldsymbol{E} \\ \boldsymbol{R}_1 \cdots \boldsymbol{R}_n \end{bmatrix} \boldsymbol{f} \quad (17)$$

where $\boldsymbol{E} \in \mathbb{R}^{3\times3}$ represents an identity matrix, $\boldsymbol{R}_i \in \mathbb{R}^{3\times3}$ indicates a cross product. From Eq.(16) and Eq.(17), $\boldsymbol{F}$ can be represented in the following span format:

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{E} \cdots \boldsymbol{E} \\ \boldsymbol{R}_1 \cdots \boldsymbol{R}_n \end{bmatrix} \boldsymbol{B}_{span} \boldsymbol{S} = \boldsymbol{A}_{span} \boldsymbol{S} \geqq 0 \quad (18)$$

Conforming to the convex polyhedron theory [30], $\boldsymbol{F}$ can be expressed in face format as follows:

$$\boldsymbol{A}_{face} \boldsymbol{F} \leqq 0 \quad (19)$$

where $\boldsymbol{A}_{face}$ represents the matrix in face format of $\boldsymbol{A}_{span}$.

REFERENCES

[1] T. Ito, K. Ayusawa, E. Yoshida, and H. Kobayashi, "Simultaneous control framework for humanoid tracking human movement with interacting wearable assistive device," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3604–3611, 2020.

[2] M. Dežman, S. Massardi, D. Pinto-Fernandez, V. Grosu, C. Rodriguez-Guerrero, J. Babič, and D. Torricelli, "A mechatronic leg replica to benchmark human–exoskeleton physical interactions," *Bioinspiration & Biomimetics*, vol. 18, no. 3, p. 036009, 2023.

[3] P. Ferrari, L. Rossini, F. Ruscelli, A. Laurenzi, G. Oriolo, N. G. Tsagarakis, and E. Mingo Hoffman, "Multi-contact planning and control for humanoid robots: Design and validation of a complete framework," *Robotics and Autonomous Systems*, vol. 166, p. 104448, 2023.

[4] A. Adu-Bredu, G. Gibson, and J. W. Grizzle, "Exploring kinodynamic fabrics for reactive whole-body control of underactuated humanoid robots," *arXiv preprint arXiv:2303.04279*, 2023.

[5] T. Inamura and Y. Mizuchi, "Sigverse: A cloud-based vr platform for research on multimodal human-robot interaction," *Frontiers in Robotics and AI*, vol. 8, 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2021.549360

[6] E. Sheetz, M. Kian, M. Savchenko, S. J. Jorgensen, E. Laske, S. Azimi, M. Mataric, and O. C. Jenkins, "Comparing human input in rviz and virtual reality interfaces for 3d object manipulation tasks," in *First International Workshop on Horizons of an Extended Robotics Reality at IEEE IROS 2022*, 2022.

[7] D. Lim, D. Kim, and J. Park, "Online telemanipulation framework on humanoid for both manipulation and imitation," in *2022 19th Int. Conf. on Ubiquitous Robots (UR)*. IEEE, 2022, pp. 8–15.

[8] P. M. Viceconte, R. Camoriano, G. Romualdi, D. Ferigo, S. Dafarra, S. Traversaro, G. Oriolo, L. Rosasco, and D. Pucci, "Adherent: Learning human-like trajectory generators for whole-body control of humanoid robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2779–2786, 2022.

[9] K. Darvish, L. Penco, J. Ramos, R. Cisneros, J. Pratt, E. Yoshida, S. Ivaldi, and D. Pucci, "Teleoperation of humanoid robots: A survey," *IEEE Trans. on Robotics*, vol. 39, no. 3, pp. 1706–1727, 2023.

[10] H. Bock and K. Plitt, "A multiple shooting algorithm for direct solution of optimal control problems," *IFAC Proceedings Volumes*, vol. 17, no. 2, pp. 1603–1608, 1984, 9th IFAC World Congress: A Bridge Between Control Science and Technology, Budapest, Hungary, 2-6 July 1984.

[11] I. Maroger, N. Ramuzat, O. Stasse, and B. Watier, "Human trajectory prediction model and its coupling with a walking pattern generator of a humanoid robot," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6361–6369, 2021.

[12] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. D. Prete, "Optimization-based control for dynamic legged robots," *IEEE Transactions on Robotics*, vol. 40, pp. 43–63, 2024.

[13] P. Besse and J. O. Ramsay, "Principal components analysis of sampled functions," *Psychometrika*, vol. 51, no. 2, pp. 285–311, 1986.

[14] H. L. Shang, "A survey of functional principal component analysis," *AStA Advances in Statistical Analysis*, vol. 98, pp. 121–142, 2014.

[15] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[16] D. Holden, J. Saito, and T. Komura, "A deep learning framework for character motion synthesis and editing," *ACM Trans. on Graphics (TOG)*, vol. 35, no. 4, pp. 1–11, 2016.

[17] T. Lesort, N. Díaz-Rodríguez, J.-F. Goudou, and D. Filliat, "State representation learning for control: An overview," *Neural Networks*, vol. 108, pp. 379–392, 2018.

[18] S. Shimizu, K. Ayusawa, and G. Venture, "Pseudo direct and inverse optimal control based on motion synthesis using fpca," in *2020 IEEE-RAS 20th Int. Conf. on Humanoid Robots*, 2021, pp. 291–298.

[19] Z. Xie, L. Li, and X. Luo, "Optimization of the ground reaction force for the humanoid robot balance control," *Acta Mechanica*, vol. 232, no. 10, pp. 4151–4167, 2021.

[20] R. Luxman and T. Zielinska, "Robot motion synthesis using ground reaction forces pattern: Analysis of walking posture," *Int. J. of Advanced Robotic Systems*, vol. 14, no. 4, p. 1729881417720873, 2017.

[21] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of honda humanoid robot," in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 2, 1998, pp. 1321–1326 vol.2.

[22] S.-H. Lee and A. Goswami, "Ground reaction force control at each foot: A momentum-based humanoid balance controller for non-level and non-stationary ground," in *2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, 2010, pp. 3157–3162.

[23] J. Colombel, D. Daney, and F. Charpillet, "On the reliability of inverse optimal control," in *2022 Int. Conf. on Robotics and Automation*, 2022, pp. 8504–8510.

[24] M. Kayano and S. Konishi, "Functional principal component analysis via regularized gaussian basis expansions and its application to unbalanced data," *J. of Statistical Planning and Inference*, vol. 139, no. 7, pp. 2388–2398, 2009.

[25] W. Ke, Y. Bai, H. Li, K. Chen, and Q. Yuan, "Control of stepping downstairs for humanoid robot based on dynamic multi-objective optimization," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 5, p. e5999, 2021.

[26] T. Matsui, M. Motegi, and N. Tani, "Mathematical model for simulating human squat movements based on sequential optimization," *Mechanical Engineering J.*, vol. 3, no. 2, pp. 15–00 377–15–00 377, 2016.

[27] K. Kaneko, F. Kanehiro, M. Morisawa, K. Akachi, G. Miyamori, A. Hayashi, and N. Kanehira, "Humanoid robot hrp-4 - humanoid robotics platform with lightweight and slim body," in *2011 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2011, pp. 4400–4407.

[28] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.

[29] K. Ayusawa and E. Yoshida, "Comprehensive theory of differential kinematics and dynamics towards extensive motion optimization framework," *The International Journal of Robotics Research*, vol. 37, no. 13-14, pp. 1554–1572, 2018.

[30] G. B. Dantzig, *Linear inequalities and related systems*. Princeton university press, 1956, no. 38.