

Whole-Body Motion Planning for Humanoid Robots by Specifying Via-Points

Regular Paper

ChangHyun Sung^{1,*}, Takahiro Kagawa¹ and Yoji Uno¹

¹ Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Japan

* Corresponding author E-mail: s_changhyun@nuem.nagoya-u.ac.jp

Received 11 Sep 2012; Accepted 11 Jun 2013

DOI: 10.5772/56747

© 2013 Sung et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract We design a framework about the planning of whole body motion for humanoid robots. Motion planning with various constraints is essential to success the task. In this research, we propose a motion planning method corresponding to various conditions for achieving the task. We specify some via-points to deal with the conditions for target achievement depending on various constraints. Together with certain constraints including task accomplishment, the via-point representation plays a crucial role in the optimization process of our method. Furthermore, the via-points as the optimization parameters are related to some physical conditions. We applied this method to generate the kicking motion of a humanoid robot HOAP-3. We have confirmed that the robot was able to complete the task of kicking a ball over an obstacle into a goal in addition to changing conditions of the location of a ball. These results show that the proposed motion planning method using via-point representation can increase articulation of the motion.

Keywords Decentralized Control, High-Order Neural Networks, Extended Kalman Filter, Backstepping

1. Introduction

Humanoid robots with many degrees of freedom can perform a variety of movements. If humanoid robots had the ability to adapt themselves to the human environment, the robots could play useful roles as substitutes for the humans in various areas such as working in construction and nursing care. However, in planning the motion for humanoid robots, the redundancy problem has to be solved under a great number of constraints, such as maintaining balance, range of motion, and torque limitation. Furthermore, the conditions for achieving the target also need to be considered in the motion plan.

Accomplishment of the task is an important factor in motion planning. Despite the fact that the motion has been planned by considering a set of constraints such as balance, the planned motion is not beneficial if the robot does not accomplish the task. To achieve the motion objective, it is important to represent the conditions for achieving the task. These conditions are usually represented by equality constraints in motion planning for the robots. By representing the appropriate equality

constraints during the motion, the robot can achieve the goal of motion.

The equation for whole body motion for humanoid robots has strong nonlinearity, since the movements are generated by a combination of multi-joints. In previous works, humanoid robots were controlled by using simple linear models, e.g., a 2D or 3D inverted pendulum model [1-2] or a cart on table model [3]. Moreover, there are motion planning methods based on specific physical quantities, e.g., resolved momentum [4], and angular momentum [5]. The motion for humanoid robots with complex kinematics and dynamics can be calculated quickly using simplified or reduced models. However, there are many constraints that should be considered, and the motion planning method introduced by these models cannot deal with these.

To resolve the above problem, motion planning can be transformed into a constrained nonlinear optimization problem by joint angle trajectory parameterization [8]. Constraints such as balance and joint range of the motion are considered as inequality constraints, while the conditions for target achievement are represented by equality constraints in the optimization process. The problem of control for a humanoid robot is solved by Quadratic Programming (QP) [6] or Second Order Cone Programming (SOCP) [7]. Moreover, Semi Infinite Programming (SIP) is used to resolve the problem of whole body motion planning for humanoid robots by joint angle trajectory parameterization [8-9]. Motion planning as SIP is useful for a variety of constraints, such as multi-contact motion [10].

However, joint angle trajectory parameterization in the motion planning as constrained optimization problem remains a critical problem to increase articulation of the motion. In previous works, joint angle trajectories have often been parameterized using B-spline basis function [11]. In this case, the motion planning problem is solved by calculating adequate coefficients of B-spline basis function [8-10]. However, the coefficients do not explicitly represent any quantities related to the motion. Therefore, even if only specific condition is changed in the re-planning, all the parameters have to be recalculated. If we define the optimization parameters as quantities related to the trajectory, we would reduce the number of optimization parameter under the change of certain conditions.

Via-point representation, which is mathematically straightforward to calculate, is a way of representing the equality constraints during the motion. The advantage of the via-point representation is that via-points as an optimization parameter is explicitly associated with some physical conditions. A via-point is a significant component in quantitative reproduction of human

exercise. Dynamic movement of a human can be expressed by assigning various via-points and calculating joint trajectories based on the principle of optimization [12-13]. Furthermore, via-point representation based on human movement is used to control the robot manipulator. By taking some via-points from measured human motion and applying them to the robot manipulators, the robot can perform the same task as a human [14].

In this research, we propose a new motion planning method using via-point representation to deal with various conditions for accomplishing tasks. Our method deals with various constraints including the conditions for achieving the target, as conditions of the via-point in the optimization process. First, we represent the joint angle trajectory of the robot by the conditions of the via-points based on a jerk cost function. Second, the motion is planned by calculating the optimal parameters of the via-points satisfied with various constraints, including the conditions for target achievement. This problem can be regarded as the SIP problem [15]. In addition, we introduce simple re-planning technique by changing some parameters. This method produces a motion corresponding to changing some conditions.

Kicking motion, which is one type of whole body motion, is performed to kick a ball into a goal. A humanoid robot needs to maintain its balance on one supporting leg, and complete the task while taking into account the changing surrounding conditions with the other flying leg. To be able to effectuate a variety of motion in robots, the motion must be planned according to the motion objective. In our proposed motion planning method, joint angle trajectories already include the conditions for target achievement as certain via-points. Therefore, the robot can deal with varying conditions for accomplishing the task. Furthermore, other constraint conditions such as maintaining balance are also integrated as the conditions of via-points. Our method is applied to generate the kicking motion of a HOAP-3 humanoid robot. We confirmed that the robot was able to success the kicking task [16], even if there is an obstacle in front of the goal. Furthermore, the robot can re-plan the motions corresponding to various conditions of kicking speed efficiently.

The paper is organized as follows. Section 2 defines the motion planning problem using via-point representation. Section 3 presents the conditions that should be considered in motion planning for humanoid robots as the parameters of the via-point. Section 4 illustrates an application of proposed method in generating the kicking motion of a HOAP-3 humanoid robot. In Section 5, we discuss our motion planning method and compare proposed method with previous works which generate kicking motions. We conclude the paper by describing

the advantages and limitations of our motion planning method using the via-points, and emphasize prospective enhancements of our method.

2. Motion Planning Problem

In this section, we explain the motion planning problem using via-point representation. The motion planning problem is usually solved by calculating the set of optimal joint angle trajectories. In our motion planning framework, a via-point representation is used to deal with the various conditions for accomplishing the task. Furthermore, each joint angle trajectory is represented by the parameters of the via-points. In other words, the motion planning problem is essentially one of finding the optimal parameters of via-points. In the subsequent subsections, we detail the particulars of the motion planning process.

2.1 Joint Angle Trajectory using Via-point Representation

Minimum jerk cost is a typical criterion to generate multi-joint motion of a human and the robot manipulators. In the mathematical model explaining human coordination of smooth arm movement, the minimum jerk is used as a biological indicator [12]. Additionally, the jerk cost function is sufficient for considering energy efficiency in the robot control [17-18]. A simple minimum jerk trajectory can be represented by a fifth order polynomial function $\theta(t)$, which has six parameters (see Appendix A).

$$\theta(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0. \quad (1)$$

The position, velocity, and acceleration at the start time ($t = 0$) are θ_0 , $\dot{\theta}_0$, and $\ddot{\theta}_0$, and those at the end time ($t = t_f$) are θ_f , $\dot{\theta}_f$, and $\ddot{\theta}_f$, respectively. Generally, the velocity and acceleration of the initial and final states are zero, because the movement is stationary at the start and end points. As a result, the fifth order polynomial function is determined by the initial and final positions. However, this simple function cannot take into account the various constraints, since this trajectory function depends only on the positions of the start and end points. To represent the various constraints, motion planning requires an alternative method.

In motion planning, the robot can accomplish the task by specifying certain equality constraints. We make use of via-point representation to consider the conditions of the various constraints, of which target achievement is the most important. These constraints are transformed into conditions of the via-points. In this case, the specification of the via-point is an essential problem. In general whole-body motion, specifying appropriate via-points is fairly difficult problem. In this paper, we deal with the motion planning problem in which appropriate via-point to achieve the task can be obviously estimated. The

constraints are represented by the positions θ_i and velocities $\dot{\theta}_i$ of the via-points at time t_i , $i = 1, \dots, m$. For $t_1 < \dots < t_m$,

$$\begin{bmatrix} \theta(t_1) - \theta_1 \\ \dot{\theta}(t_1) - \dot{\theta}_1 \\ \vdots \\ \theta(t_m) - \theta_m \\ \dot{\theta}(t_m) - \dot{\theta}_m \end{bmatrix} = \mathbf{0}. \quad (2)$$

The joint angle trajectory $\theta(t)$ with the constraints of the m via-points can be calculated for optimization with Lagrange multipliers, $\pi_{i,1}$ and $\pi_{i,2}$ ($i = 1, \dots, m$) based on the jerk cost function (see Appendix B). We obtain following equation.

$$\theta(t) = \begin{cases} a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 & (t_0 \leq t \leq t_1) \\ a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 & (t_1 \leq t \leq t_2) \\ + g_1(t) & \vdots \\ a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 & (t_m \leq t \leq t_f) \\ + g_1(t) + g_2(t) + \dots + g_n(t) & \end{cases} \quad (3)$$

$$g_i(t) = \frac{1}{24} \pi_{i,2} (t - t_i)^4 + \frac{1}{120} \pi_{i,1} (t - t_i)^5$$

where t_1, \dots, t_m are the times at the via-points, and t_f is the end time of the movement. This equation has $6 + 2m$ parameters, determined by the six conditions of the initial and final states, and $2m$ conditions of each joint angle and angular velocity at the m via-points. We assume that the time at the via-points and the initial and final positions are given. Therefore, we need to determine the $2m$ conditions of the via-points (θ_i and $\dot{\theta}_i$, $i = 1, \dots, m$) to represent the joint angle trajectory. Briefly, we express the parameters of the via-points as

$$\begin{aligned} \mathbf{t}_{via} &= \{t_1, \dots, t_m\} \\ \boldsymbol{\theta}_{via} &= \{\theta_1, \dots, \theta_m\} \\ \dot{\boldsymbol{\theta}}_{via} &= \{\dot{\theta}_1, \dots, \dot{\theta}_m\}. \end{aligned} \quad (4)$$

The motion planning problem in this paper involves finding suitable values for the via-point parameters in Eq. (4). Here, parameters $\boldsymbol{\theta}_{via}$ and $\dot{\boldsymbol{\theta}}_{via}$ are combined and represented by \mathbf{X} .

$$\mathbf{X} = \{\boldsymbol{\theta}_{via}, \dot{\boldsymbol{\theta}}_{via}\}. \quad (5)$$

It is assumed that the number of controlled joint motors is n . There are $2mn$ parameters of the via-points that need to be specified to represent the joint angle trajectories. \mathbf{X} represents the position, posture and velocity of the humanoid robot at the given time through the via-points. Therefore, we can deal with the conditions for achieving the task as the via-point. Additionally, the set of joint angle, angular velocity and angular acceleration trajectory functions are given by

$$\begin{aligned} [\theta_1(\mathbf{X}, t) \quad \dots \quad \theta_n(\mathbf{X}, t)]^T &= \boldsymbol{\theta}(\mathbf{X}, t) \\ [\dot{\theta}_1(\mathbf{X}, t) \quad \dots \quad \dot{\theta}_n(\mathbf{X}, t)]^T &= \dot{\boldsymbol{\theta}}(\mathbf{X}, t) \\ [\ddot{\theta}_1(\mathbf{X}, t) \quad \dots \quad \ddot{\theta}_n(\mathbf{X}, t)]^T &= \ddot{\boldsymbol{\theta}}(\mathbf{X}, t), \end{aligned} \quad (6)$$

where t denotes an arbitrary time in the interval $[0, t_f]$.

2.2 Motion Planning as a SIP Problem

We can determine the joint angle trajectories by optimizing parameter \mathbf{X} of the via-points. However, there are numerous constraints during the motion that have to be considered. In these constraints, the conditions that are satisfied over the whole motion duration, e.g., balance and range of motion, are represented by inequality constraints. The function of the constraints that are represented by inequality can be expressed by the joint angle, angular velocity, and angular acceleration functions. The function of the inequality constraints is expressed as

$$\tilde{c}(\theta(\mathbf{X}, t), \dot{\theta}(\mathbf{X}, t), \ddot{\theta}(\mathbf{X}, t)) = c(\mathbf{X}, t) \leq 0. \quad (7)$$

Moreover, we can define the function of the inequality constraints using only the set of joint angle parameters \mathbf{X} and time t , because the joint angle, angular velocity, and angular acceleration trajectories are determined by \mathbf{X} and t .

We consider the constraints to succeed in accomplishing the task directly using via-point representation. The conditions for target achievement represented by equality constraints need to be considered only at time t_{via} of the via-points. Therefore, we can represent the conditions for achieving the task as equality constraints using only \mathbf{X} .

$$\tilde{ceq}(\theta(\mathbf{X}, t_{via}), \dot{\theta}(\mathbf{X}, t_{via}), \ddot{\theta}(\mathbf{X}, t_{via})) = ceq(\mathbf{X}) = 0. \quad (8)$$

We focus on the nonlinear constrained optimization problem to determine parameter \mathbf{X} of the joint angle trajectory. Indeed, the motion planning problem is transformed into a SIP problem [15], which consists of finding parameter \mathbf{X} of the joint angle trajectory.

$$\begin{aligned} &\text{argmin}_{\mathbf{X}} \quad C(\mathbf{X}) \\ &\text{s.t.} \quad c(\mathbf{X}, t) \leq 0, ceq(\mathbf{X}) = 0 \\ &\quad \quad \forall t \in [0, t_f], \end{aligned} \quad (9)$$

where $C(\mathbf{X})$ is cost function to solve redundancy problem. Specifically, various constraints are considered by the states of via-points in this motion planning problem.

2.3 Partial Re-planning as a SIP Problem

An advantage of our method is that the optimization parameters consist of the kinematic quantities related to the motion of the robot. Indeed, θ_{via} and $\dot{\theta}_{via}$ are related to the conditions of position and velocity, respectively. Therefore, we can deal with only certain conditions without optimizing all parameters. For example, it is possible to suppress only conditions of velocity for achieving the task. In this case, we have to recalculate only $\dot{\theta}_{via}$ under the same θ_{via} . It is not necessary to optimize entirety of \mathbf{X} . By using the calculated conditions θ_{via} , re-planning problem could be formulated as follows;

$$\begin{aligned} &\text{argmin}_{\dot{\theta}_{via}} \quad C'(\dot{\theta}_{via}) \\ &\text{s.t.} \quad c'(\dot{\theta}_{via}, t) \leq 0, ceq'(\dot{\theta}_{via}) = 0, \end{aligned} \quad (10)$$

where $C'(\dot{\theta}_{via})$, $c'(\dot{\theta}_{via}, t)$ and $ceq'(\dot{\theta}_{via})$ are new cost function, inequality and equality constraint functions. When some conditions to accomplish the task are changed, the robot can re-plan a motion by optimizing some of the via-point parameters.

3. Motion Planning for Humanoid Robots

In this section, we describe motion planning for humanoid robots using the problem defined in Section 2. In the following, we illustrate the conditions that must be considered in motion planning for humanoid robots.

3.1 Humanoid Robots

First, we represent the kinematics, dynamics, and balance of humanoid robots with parameter \mathbf{X} of the joint angle trajectory.

3.1.1 Kinematics

Once the joint angle trajectories have been determined, the humanoid robot's joint positions, postures, velocities, and angular velocities can be calculated by the kinematics equations. The positions \mathbf{P} , postures \mathbf{R} , velocities \mathbf{V} , and angular velocities \mathbf{W} at time t are defined by parameter \mathbf{X} of the joint angle trajectory as follows;

$$\begin{aligned} \mathbf{P} &= \tilde{\mathbf{p}}(\theta(\mathbf{X}, t)) \\ \mathbf{R} &= \tilde{\mathbf{r}}(\theta(\mathbf{X}, t)) \\ \mathbf{V} &= \tilde{\mathbf{v}}(\theta(\mathbf{X}, t), \dot{\theta}(\mathbf{X}, t)) \\ \mathbf{W} &= \tilde{\mathbf{w}}(\theta(\mathbf{X}, t), \dot{\theta}(\mathbf{X}, t)) \end{aligned} \quad (11)$$

where \mathbf{P} , \mathbf{R} , \mathbf{V} , and \mathbf{W} are 3-dimensional vectors.

In motion planning, the robot can accomplish the task by constraining the kinematics information. In particular, the kinematics information at the via-points is the most important. Therefore, the kinematics information for accomplishing the task needs to be focused on the via-points. Because parameter \mathbf{X} of the via-points consists of the joint angle θ_{via} , and the angular velocity $\dot{\theta}_{via}$ at the via-points, the kinematics information at time t_{via} of the via-points is represented only by \mathbf{X} .

$$\begin{aligned} \mathbf{P}_{via} &= \mathbf{p}(\mathbf{X}), \quad \mathbf{R}_{via} = \mathbf{r}(\mathbf{X}) \\ \mathbf{V}_{via} &= \mathbf{v}(\mathbf{X}), \quad \mathbf{W}_{via} = \mathbf{w}(\mathbf{X}) \end{aligned} \quad (12)$$

These are calculated by the robot manipulator's forward kinematics [19].

3.1.2 Dynamics

We must consider each motor torque τ_j , $j = 1, \dots, n$, in motion planning owing to the upper and lower limitations in motor torque. These are calculated by the

dynamics equations. The joint angle torque can be calculated by Newton-Euler's formula [19], as expressed by the following functions:

$$\begin{bmatrix} \tau_1(t) & \cdots & \tau_n(t) \end{bmatrix}^T = \tilde{\tau}(\theta(X, t), \dot{\theta}(X, t), \ddot{\theta}(X, t)) = \tau(X, t). \quad (13)$$

3.1.3 Balance

The humanoid robot must maintain its balance during motion. The balance of a humanoid robot can be evaluated by the ZMP (Zero Moment Point) [20]. The ZMP is the point at which the total inertia force is equal to 0 on the supporting area. If the ZMP of the planned motion exists at the base of the support with sufficient margins, the humanoid robot can maintain its balance. The function of the ZMP is defined as

$$\begin{bmatrix} x_{zmp}(t) & y_{zmp}(t) \end{bmatrix}^T = \tilde{z}(\theta(X, t), \dot{\theta}(X, t), \ddot{\theta}(X, t)) = z(X, t). \quad (14)$$

where x_{zmp} and y_{zmp} are the ZMP values in the x and y directions.

3.2 Motion Planning for Humanoid Robots as a SIP Problem

Motion for humanoid robots is performed by determining parameter X of the viapoints. The problem of calculating parameter X of the joint angle trajectory is considered as a SIP problem, in which we need to set up the conditions including the cost function, and inequality and equality constraints. In the following, we describe the conditions for calculating the optimal parameters of the via-points for motion planning.

3.2.1 Inequality Constraints

The conditions for maintaining balance, and the limitations of the joint angle and torque must be satisfied throughout the duration of the whole motion. These constraints can be represented by inequality constraints in the SIP problem.

$$\begin{aligned} Z^- &\leq z(X, t) \leq Z^+ \\ \theta^- &\leq \theta(X, t) \leq \theta^+ \\ T^- &\leq \tau(X, t) \leq T^+ \end{aligned} \quad (15)$$

Here, (Z^-, Z^+) , (θ^-, θ^+) , and (T^-, T^+) are the lower and upper limits on the ZMP, joint angle, and torque, respectively. The dimensions of vectors Z , θ , and T are, respectively, 2, n , and n , where n is the number of controlled joints. Specifically, support area of ZMP is simplified as a region within the inscribed rectangular area in the foot to consider x_{zmp} and y_{zmp} separately in optimization.

Furthermore, the range of motion, such as the positions and postures of the joints, needs to be restricted. The limits of the joint positions and postures can also be represented by inequality constraints:

$$\begin{aligned} P^- &\leq p(X, t) \leq P^+ \\ R^- &\leq r(X, t) \leq R^+ \end{aligned} \quad (16)$$

Here, $p(X, t)$ and $r(X, t)$ are the kinematics functions for calculating the positions and postures of the joint, and (P^-, P^+) and (R^-, R^+) are the lower and upper limits on the positions and postures of the joints, respectively. The inequality constraint function represented by Eq. (7), includes Eqs. (15) and (16) in the motion planning for humanoid robots.

3.2.2 Equality Constraints

The conditions for achieving the goal during the motion are represented by equality constraints. We assume that the target information for accomplishing the task is given by the position P_d , posture R_d , velocity V_d , and angular velocity W_d of the joints. Humanoid robots can accomplish the task by performing the motion as given by the target information. In our study, we use via-point representation to deal with the conditions for achieving the target. In our method, only the state at the via-points is considered to be the target information. Therefore, the conditions for target achievement are expressed only using only the parameter X of the via-point as follows:

$$\begin{aligned} p(X) - P_d &= 0, & r(X) - R_d &= 0 \\ v(X) - V_d &= 0, & w(X) - W_d &= 0 \end{aligned} \quad (17)$$

Finally, the conditions for accomplishing the task given in Eq. (17) are represented by the equality constraint function given in Eq. (8).

3.2.3 Cost Function

The cost function $C(X)$ is the most important in the optimization process for motion planning. The characteristic of robot motion depends on the choice of the cost function. The cost function, e.g., minimizing the sum of squared torque [9] and energy consumption [9], can be sufficient in improving the autonomy of the robot. Biologically inspired cost functions such as the minimum torque change [13] can also be considered. Minimized jerk [12,17] is a typical cost function for robot manipulators, because the criterion requires a smooth trajectory for adequate motor control. In this research, the joint angle trajectory with the via-points is already optimized using a minimized jerk cost function. In other words, the trajectory is optimal under the conditions that are satisfied with the constraints as the via-points. For this reason, to determine the joint angle trajectory's parameter X , we set up a criterion function to improve the motion performance. The details are explained in Section 4 through application to generate the kicking motion for humanoid robots.

4. Application to Kicking Motion

In this section, we describe a simple application of the proposed motion planning method. We applied our

motion planning method to generate the kicking motion for a HOAP-3 humanoid robot. Kicking motion is one of the motions in which explicit via-point can be easily estimated; e.g., the moment when robot kicks the ball. Various methods based on optimization have been developed to generate kicking motion for humanoid robots [4,8-10]. Although these methods considered the constraints of balance, and ranges of joint angles and torques, they did not address various requirements to perform the task depending on the surrounding environment.

On the other hand, Xu et al. proposed a control-based planning method for kicking motion [21]. It enables the robot to perform an online motion control with fast planning technique of kicking trajectory. In their method, the joint angle trajectories were generated by inverse kinematics. However, a redundancy problem had to be solved in the use of the inverse kinematics equations. Furthermore, singular posture problem had to be taken into account for the planning. To avoid these problems, it is desired to plan the motions in joint space.

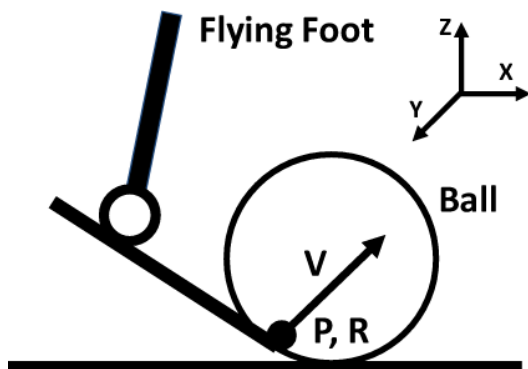


Figure 1. The moment when a humanoid robot kicks a ball. (*P*: position, *R*: posture, *V*: velocity)

Nevertheless, these studies set out to consider how to kick a ball, which imposed certain conditions for accomplishing the task in the motion planning. Indeed, in previous works, the robot always kicked a ball on the ground, even if the robot had the ability to respond when the location of the ball changed [9,21]. For example, if there is an obstacle in front of the robot, the robot needs to kick the ball in the air to score a goal [16]. As such, we must consider the course of the ball, including how to kick it.

In the kicking motion, the most important point is the instant at which the robot kicks the ball. A humanoid robot can perform a variety of kicking motions depending on the surrounding conditions by considering the information when the robot kicks the ball. We improve the performance of the kicking motion by constraining the information at the instant when the robot kicks the ball by means of a via-point. Here, we use only one via-point which corresponds to the moment when the

robot kicks the ball. Based on Eq. (3), the joint angle trajectory with one via-point is expressed as (see Appendix B)

$$\theta(t) = \begin{cases} a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 & (t_0 \leq t \leq t_{via}) \\ a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 + \frac{1}{24} \pi_2 (t - t_1)^4 + \frac{1}{120} \pi_1 (t - t_1)^5 & (t_{via} \leq t \leq t_f) \end{cases} \quad (18)$$

Eq. (18) has eight parameters ($a_0, a_1, a_2, a_3, a_4, a_5, \pi_1, \pi_2$), which are determined by six conditions namely angle, angular velocity ($= 0$), and angular acceleration ($= 0$) at the start and end points, and two further conditions, namely, the angle(θ_{via}) and angular velocity($\dot{\theta}_{via}$) at the via-point. In generating the kicking motion, we specify that the initial and final positions are the standing state. Furthermore, the time at the via-point at which the robot kicks the ball is given. As a result, we need to determine the two conditions of the via-point to represent joint angle trajectory. These parameters of the via-point are represented as \mathbf{X} . In our application to kicking motion, we focus on how the robot kicks the ball in addition to how the robot responds to the ball changing location. In this paper, we introduce two types of kicking motion, that is, kicking a ball on the ground and in the air.

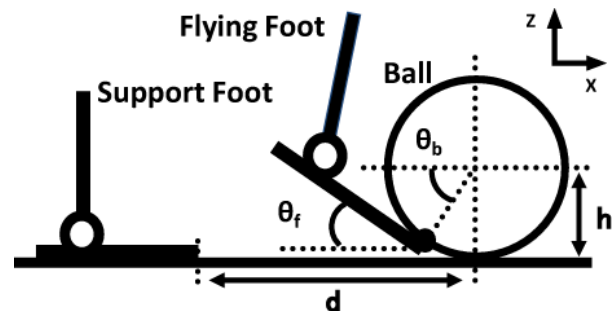


Figure 2. Experimental environment for kicking motion. (*d*: distance from the center of the ball to the toe of the supporting foot; *h*: height of the center of the ball; θ_f : angle of the flying foot relative to the horizontal axis; θ_b : angle of the line between the center of the ball and the contact point.)

4.1 Planning of Kicking Motion for HOAP-3 Humanoid Robot

We use a HOAP-3 humanoid robot with 28 degrees of freedom, height 60 [cm], and weight 8.8 [kg]. Controlled joints in the generation of the kicking motion are the lower body's 12 motors and upper body's 1 motor. In total, 13 motors must be controlled in generating the kicking motion (except only arm motors). Therefore, the number of parameters of the joint angle trajectory \mathbf{X} is 26 (2×13 , 2 conditions of the via-point and 13 controlled joint motors).

The instant when the robot kicks the ball is considered as to be time t_{via} of the via-point. We decided that the motion to kick the ball would be performed by the right foot. By determining parameter \mathbf{X} of the joint angle trajectory, we can calculate position \mathbf{P} , posture \mathbf{R} , and

velocity \mathbf{V} of the flying foot's toe at the moment when the robot kicks the ball (Fig. 1) by using the forward kinematics equation (origin: left foot's toe, end effector: right foot's toe). For simplicity, we ignore the angular velocity \mathbf{W} when the robot kicks a ball. The robot can accomplish the kicking task taking into account the surrounding conditions by controlling the information of the moment when the robot kicks the ball. In the following, we describe the generation of the kicking motion for the HOAP-3 by setting up the cost function, and inequality and equality constraint functions of the SIP problem. In this research, *fmincon* function of a numerical computing language MATLAB supplied by Math Works Inc. was used to solve the SIP Problem.

4.1.1. Inequality Constraint Function

The inequality constraint function represents the conditions that must be satisfied during the movement. In generating a kicking motion, we must deal with the conditions of the limits of each segment's angle and torque, maintaining balance, and collision avoidance with the ground.

$$\begin{aligned} \theta^- &\leq \theta(\mathbf{X}, t) \leq \theta^+ \\ T^- &\leq \tau(\mathbf{X}, t) \leq T^+ \\ 0 &\leq p_z(\mathbf{X}, t) \\ \mathbf{Z}^- + \alpha_z^- &\leq \mathbf{z}(\mathbf{X}, t) \leq \mathbf{Z}^+ + \alpha_z^+ \end{aligned} \quad (19)$$

The lengths of (θ^-, θ^+) , (T^-, T^+) , and $(\mathbf{Z}^-, \mathbf{Z}^+)$ are 13, 13, and 2, respectively. $p_z(\mathbf{X}, t)$ is the vertical position of the flying foot that prevents collision with the ground. For the ZMP, if $\mathbf{z}(\mathbf{X}, t)$ has sufficient margins (α_z^-, α_z^+) for the boundary line in the supporting area $(\mathbf{Z}^-, \mathbf{Z}^+)$, the robot can maintain its balance. The margins of ZMP are set by trial and error. Finally, the inequality constraint function including Eq. (19) is represented by

$$\mathbf{c}(\mathbf{X}, t) = \begin{bmatrix} \theta(\mathbf{X}, t) - \theta^+ \\ -\theta(\mathbf{X}, t) + \theta^- \\ \tau(\mathbf{X}, t) - T^+ \\ -\tau(\mathbf{X}, t) + T^- \\ -p_z(\mathbf{X}, t) \\ \mathbf{z}(\mathbf{X}, t) - (\mathbf{Z}^+ + \alpha_z^+) \\ -\mathbf{z}(\mathbf{X}, t) + (\mathbf{Z}^- + \alpha_z^-) \end{bmatrix} \leq \mathbf{0} \quad (20)$$

4.1.2 Equality Constraint Function

The equality constraint function deals with the conditions for accomplishing the task. Indeed, the toe position \mathbf{P}_d and posture \mathbf{R}_d of the flying foot when the robot kicks a ball are constrained by the equality constraint function.

$$\mathbf{p}(\mathbf{X}) - \mathbf{P}_d = \mathbf{0}, \mathbf{r}(\mathbf{X}) - \mathbf{R}_d = \mathbf{0} \quad (21)$$

The size of each of the vectors \mathbf{P}_d and \mathbf{R}_d is three. We assume that the information for \mathbf{P}_d and \mathbf{R}_d is given. Therefore, we can express the equality constraint function as

$$\mathbf{ceq}(\mathbf{X}) = \begin{bmatrix} \mathbf{p}(\mathbf{X}) - \mathbf{P}_d \\ \mathbf{r}(\mathbf{X}) - \mathbf{R}_d \end{bmatrix} = \mathbf{0} \quad (22)$$

The humanoid robot can accurately kick a ball with the given conditions for \mathbf{P}_d and \mathbf{R}_d .

4.1.3 Cost Function

The velocity \mathbf{V}_d of the flying foot's toe when the robot kicks a ball is the most important condition in accomplishing the kicking task. In our motion planning, a cost function is used to improve the motion performance. In the application to generate the kicking motion, the velocity when the robot kicks a ball is optimized by the cost function depending on the situation.

	Exp. 1 (Kicking a ball on the ground)	Exp. 2 (Kicking a ball in the air)
Movement Time (t_f)	1.5 [sec]	
Impact Time (t_{via})	0.8 [sec]	
Posture of Flying Foot ((θ_f, θ_b))	(0, 0) [deg]	(45, 45) [deg]
Location of the Ball (d, h)	1-a: (6.5, 3.5) [cm] 1-b: (8.5, 3.5) [cm] 1-c: (4.5, 3.5) [cm]	2-a: (6.5, 3.5) [cm] 2-b: (8.5, 3.5) [cm] 2-c: (4.5, 3.5) [cm]
Cost Function	$\max V_x$	$\max 2V_x V_z / g$
Optimized Velocity* (V_x, V_z, V_z)	1-a: (2.09*, 0, 0) [m/s] 1-b: (1.83*, 0, 0) [m/s] 1-c: (2.29*, 0, 0) [m/s]	2-a: (2.13*, 0, 0.38*) [m/s] 2-b: (2.02*, 0, 0.42*) [m/s] 2-c: (2.20*, 0, 0.36*) [m/s]
Computation Time (Step Size: 5 [ms])	1-a: 572.05 [sec] 1-b: 612.27 [sec] 1-c: 950.52 [sec]	1-a: 503.95 [sec] 1-b: 259.93 [sec] 1-c: 291.81 [sec]

Table 1. Experimental environment and Computation time.

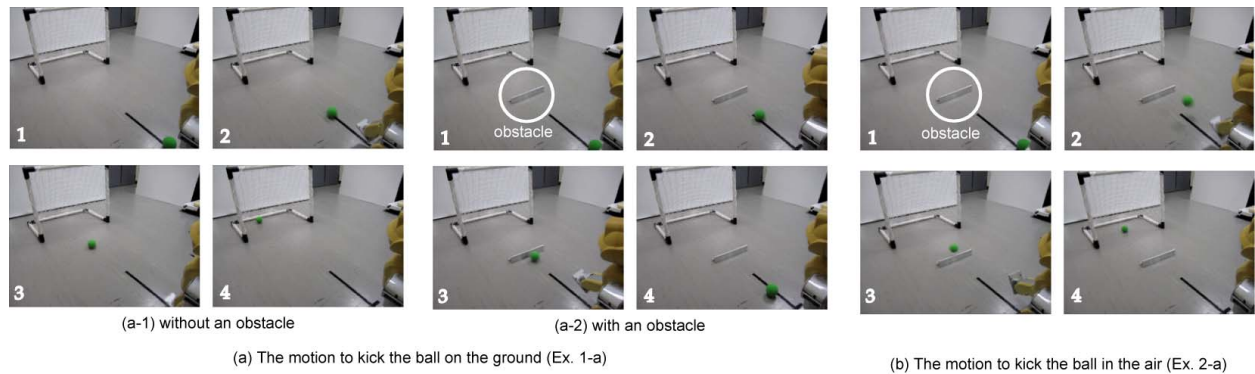


Figure 3. Experimental scenes showing two kinds of kicking motion by the HOAP-3 humanoid robot.

In kicking a ball on the ground, for example, we can improve the motion performance by minimizing the time to reach the goal. To minimize the time to reach the goal, the robot has to kick the ball with maximum velocity in the horizontal direction under the various constraints such as balance. We can represent the criterion for maximizing the horizontal velocity of the flying foot when the robot kicks a ball as

$$\max V_x \quad (23)$$

Here V_x is the horizontal velocity of the flying foot's toe when the robot kicks a ball. In generating motion to kick a ball on the ground, the velocity of the flying foot in the other directions, V_y and V_z , is set to 0.

To kick a ball in the air, we need to consider the velocity in both the horizontal and vertical directions (x and z directions). To improve the performance in kicking a ball in the air, we can consider maximizing the flying distance or highest reached point. In the current work, we use a cost function to maximize the flying distance. If the ball can be assumed to be a mass point and the rotation of the ball can be ignored, the cost function for maximizing flying distance is expressed as

$$\max 2V_x V_z / g \quad (24)$$

where g is gravitational acceleration.

4.2 Experiments involving Kicking Motion for the HOAP-3

Two types of kicking motion for a HOAP-3 humanoid robot were generated using our proposed method with the target information given. The location of the ball (d and h) and the kicking point (θ_f) and posture (θ_b) are illustrated in Fig. 2. Position P_d and posture R_d of the flying foot (Fig. 1) were calculated by d , h , θ_f and θ_b .

Table 1 shows the experimental environment for the investigation of the kicking motion.

Fig. 3 shows scenes of the kicking motion of the HOAP-3 humanoid robot corresponding to the parameter values for experiments 1 and 2 as given in Table 1. Fig. 3a shows the robot kicking a ball on the ground. In Fig. 3a, location of the ball is placed 6.5 [cm] away (Exp. 1-a). The robot was able to score a goal by kicking the ball on the ground, if there was no obstacle (Fig. 3a-1). However, with an obstacle in front of the goal, the ball did not reach the goal when kicked on the ground (Fig. 3a-2). In addition, the HOAP-3 succeeded in kicking a ball in the air as shown in Fig. 3b. The situation shown in Fig 3b is also with the ball located 6.5 [cm] distant from the robot (Exp. 2-a). We confirmed that kicking the ball in the air, the ball flew in a parabola through the air and reached the goal over the obstacle. Therefore, the robot can accomplish the kicking task irrespective of the surrounding conditions.

Fig. 4 illustrates the planned trajectories of the flying foot for the two types of kicking motion. We can see that the robot took reciprocatory action by setting only one via-point. When kicking a ball in the air, the velocity in the z -direction at the via-point is higher than when kicking a ball on the ground. Therefore, the peak of the vertical position when kicking a ball in the air (Fig. 4b) is higher than that when kicking a ball on the ground (Fig. 4a).

Additionally, the posture of the flying foot was horizontal and slanted in Figs. 4a and 4b. To kick a ball in the air, the robot angled the flying foot and aimed the kick at the lower point of the ball. The difference in the trajectories of the flying foot is the result of the given information only at the via-point.

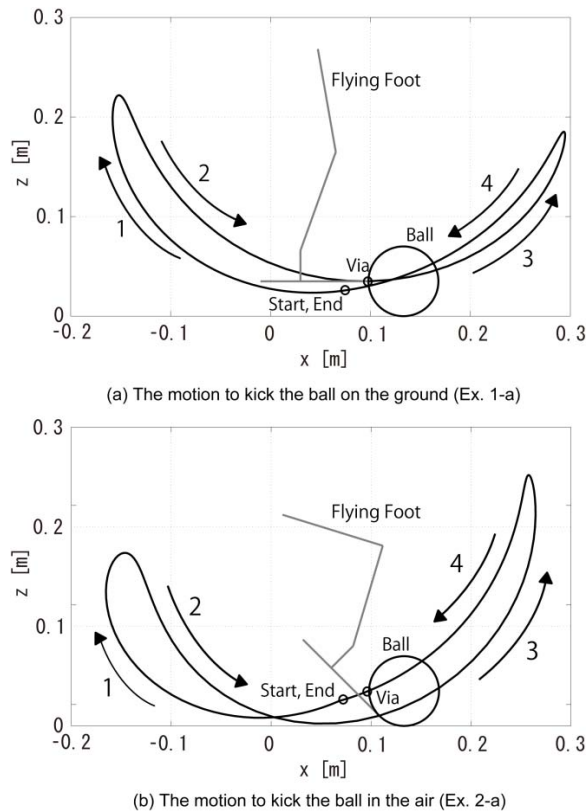


Figure 4. Planned trajectory of the flying foot on the x-z plane in Exp. 1-a and 2-a.

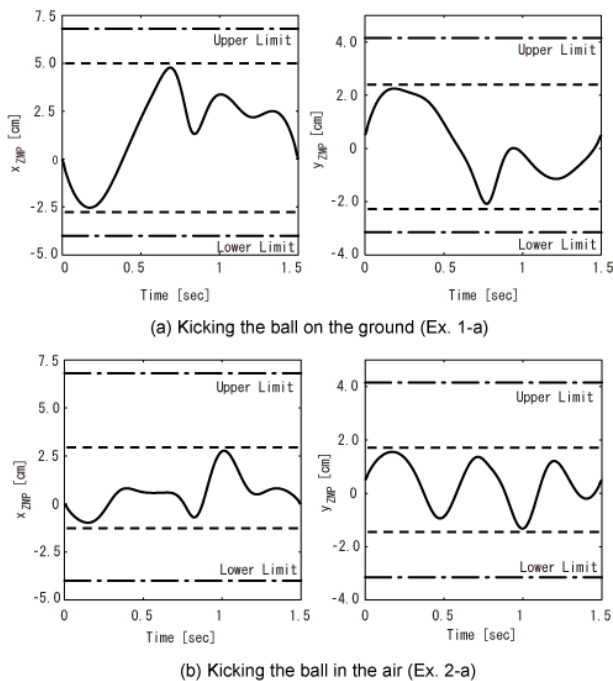


Figure 5. Planned ZMP of an HOAP-3 humanoid robot for two kinds of kicking motion. (Vertical axis of the left and right graphs in each subfigure: ZMP of the x- and y-axes, respectively; Horizontal axis of all graphs: time; Dash-dotted lines: upper and lower limits of supporting foot; Dashed lines: upper and lower limits of optimization process.)

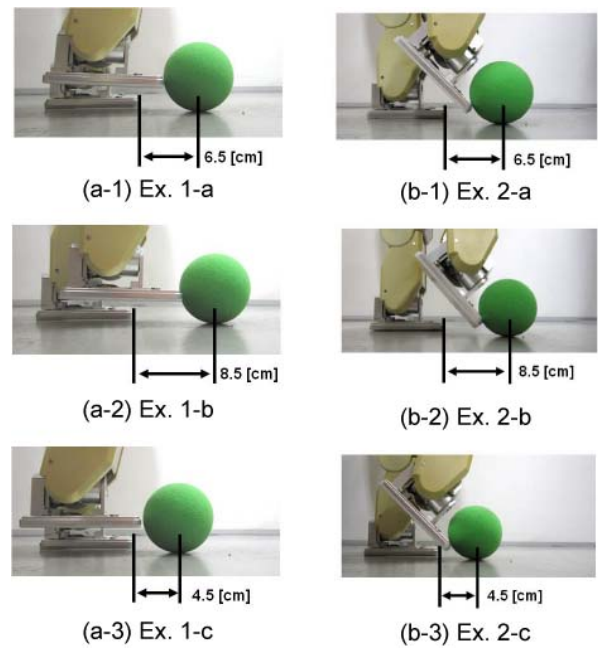


Figure 6. Images of the kicking motion of an HOAP-3 at the via-point.

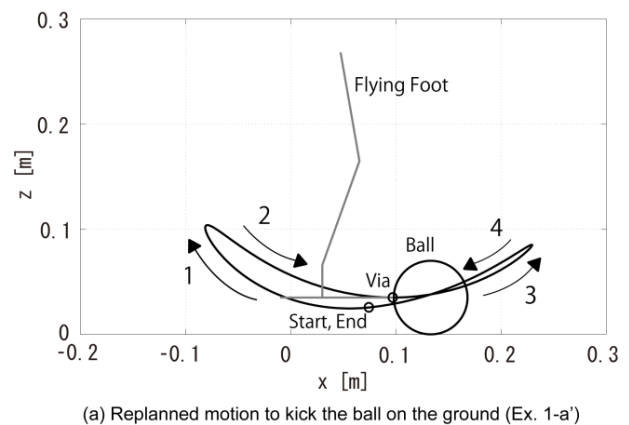


Figure 7. Replanned trajectory of the flying foot in Exp. 1-a'.

The planned ZMP of the HOAP-3 humanoid robot is shown in Fig. 5. The upper and lower boundaries on the supporting area of the x and y directions are $(6.8 [cm], -4 [cm])$ and $(4.15 [cm], -3.15 [cm])$, respectively. With sufficient margins (between the dash-dotted line and dashed line in Fig. 5) for the upper and lower limits, the HOAP-3 was able to maintain its balance during each kicking motion.

Furthermore, the location of the ball was varied in Exp. 1 (a: $6.5 [cm]$, b: $8.5 [cm]$, c: $4.5 [cm]$) and Exp. 2 (a: $6.5 [cm]$, b: $8.5 [cm]$, c: $4.5 [cm]$). Fig. 6 shows the moment at which the robot kicked the ball. Despite the location of the ball and also posture of the robot being changed, the robot could deal with all situations using the proposed planning method. We also confirmed that optimized velocities V (Fig. 1) when the robot kicked the ball in Table 1.

	Exp. 1' (Kicking a ball on the ground)	Exp. 2' (Kicking a ball in the air)
Optimized Velocity* (V_x, V_y, V_z)	1-a': (1.13 *, 0, 0) [m/s] 1-b': (0.94*, 0, 0) [m/s] 1-c': (1.10*, 0, 0) [m/s]	2-a': (1.05*, 0, 0.19*) [m/s] 2-b': (1.01*, 0, 0.21*) [m/s] 2-c': (1.09*, 0, 0.18*) [m/s]
Computation Time (Step Size: 5 [ms])	1-a': 99.51 [sec] 1-b': 67.56 [sec] 1-c': 79.32 [sec]	1-a': 91.64 [sec] 1-b': 93.12 [sec] 1-c': 159.55 [sec]

Table 2. Calculation result of re-planning for changed kicking speed.

5. Discussion

In comparison with the previous works of generating kicking motions [9,21], the advanced articulation of kicking motions was implemented in the proposed method; the robot could take into account how to kick the ball such as kicking the ball in the air. However, the proposed method only provides off-line planning because of computation time. We could confirm the specific computation time of each motion from Table 1.

In the kicking motion, it is important to control kick strength. In Exp. 1' and Exp. 2', we consider the motion planning to kick the ball weaker (about half) than velocity V of Exp. 1 and Exp. 2. Table 2 shows the specific values of optimized new velocity. In this motion planning, only the conditions of V were changed. The other conditions are the same as Table 1. Fig. 7 shows the planned trajectories of the flying foot based on Eq. (10) under the same conditions of θ_{via} in Exp. 1-a. We confirmed that the trajectory was re-planned under the same postures of flying foot as Exp. 1-a.

In our motion planning method based on via-point representation, the optimization parameters have a clear physical meaning related to the trajectory, specifically joint angle and angular velocity in this research. Indeed, joint angle is directly related to position and posture of the robot and joint angular velocity is connected to velocity of the robot. Therefore, we can control the trajectory focused on certain conditions with fewer optimization parameters. In this research, we consider the situation to control kick strength. Changing only some parts of optimization parameters, we can also reduce the computation time. We can confirm that specific computation time from Table 1 and 2. In the case of Exp. 1-a to Exp. 1-a', the computation time was reduced from 572.05 [sec] to 99.51 [sec].

On the other hand, when the B-spline basis function is used for the motion planning, all parameters have to be recalculated even if the situation like Exp. 1-a'. Furthermore, the joint angle trajectory is already optimized solution under a jerk cost function. Consequently, when we use proposed method, the smoothness of the trajectory is guaranteed than using B-spline basis function.

We expect that the proposed method would be applied to plan various whole-body motions reflecting surrounding conditions; sit-to-stand motion, throwing motion, walking motion and so on. In each motion, there are important points to achieve the task; the point of seat-off in sit-to-stand motion, the releasing point of the ball in throwing motion and the time of toe-off in walking motion. These points could be regarded as the via-points to accomplish the task in the motion planning for humanoid robots.

6. Conclusion

In this paper, we focused on varying conditions for accomplishing the task in motion planning for humanoid robots. In our proposed motion planning method, we represent the joint angle trajectory using via-point representation based on a cost function minimizing jerk. The various constraints including the conditions for accomplishing the task are considered by the conditions of the via-point. The robot's motion is planned by optimizing the conditions of the via-point as a SIP problem. Furthermore, we demonstrated a simple application of the proposed method to implement kicking motion of a humanoid.

In the application to the kicking motion, the moment at which the robot kicks a ball is constrained as a via-point. Briefly, the kicking motion of a HOAP-3 humanoid robot is planned by calculating the joint angle and angular velocity at the moment when the robot kicks the ball. We successfully generated two types of kicking motion, namely, kicking a ball on the ground and in the air. The HOAP-3 can accomplish the kicking task by kicking a ball in the air, even if there is an obstacle in front of the robot. As such, we improved the kicking motion performance using our proposed method. If we consider specific external impulse model of the ball [22], the trajectory of the ball could be controlled precisely. Furthermore, the situation to control kick strength was considered. We did not have to recalculate all parameters, because optimization parameters are related to the physical conditions.

However, the computation time for optimizing a via-point is still too expensive for application to on-line

control of humanoid robots. However, the motion for robots must be planned in real-time in order to be effective in an unpredictable human environment. On-line motion planning is an essential problem for improving motion performance. In the future, we intend approaching on-line motion planning using via-point representation. In our proposed method, the parameters of the joint angle trajectory denote the joint angle and angular velocity at the via-point. These parameters can be calculated by the manipulator's constrained motion planning method. Considering these observations, we will investigate an on-line motion planning framework with via-point representation for humanoid robots.

In this research, we assumed that the number of via-points and its timing is already given. For the motions where we cannot specify the via-points such as dancing motion, it is difficult to apply the proposed method. We will expand our motion planning algorithm so as to determine the number of via-point and its timing for complex whole-body motions. We are expecting that motion performance would be improved by specifying several via-points which are important for achieving the task.

7. Acknowledgements

The authors would like to thank Associate Prof. Kouichi Taji for his helpful suggestions. This study was supported by a Grant-in-Aid for Scientific Research (B) No.21300092 and (C) No.23560526.

8. Appendix

8.1 Appendix A: Joint Angle Trajectory without Via-point Constraints

In this paper, joint angle trajectory is formulated based on cost function for minimizing jerk. First, we set the criterion to be minimized in the time interval $t \in [0, t_f]$.

$$C = \frac{1}{2} \int_0^{t_f} \left(\frac{d^3 \theta(t)}{dt^3} \right)^2 dt. \quad (A1)$$

Defining a scalar function $L = \frac{1}{2} (\ddot{\theta}(t))^2$, the optimal solution can be expressed by Euler-Lagrange equation.

$$\frac{d^3}{dt^3} \left(\frac{\partial L}{\partial \ddot{\theta}} \right) = \frac{d^6 \theta}{dt^6} = 0. \quad (A2)$$

Therefore, simple minimum jerk trajectory is represented by a fifth order polynomial.

$$\theta(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0. \quad (A3)$$

8.2 Appendix B: Joint Angle Trajectory with Via-point Constraints

For simplicity, we formulate the minimum jerk trajectory with one via-point constraint. It is easy to extend the formula

to the minimum jerk trajectory with m via-point constraints. In this paper, the constraint is given by specifying the position θ_1 and velocity $\dot{\theta}_1$ at intermediate time t_1 .

The interior boundary condition of a trajectory $\theta(t): t_0 \leq t \leq t_f$ is represented by

$$N(\theta(t_1)) = \begin{bmatrix} \theta(t_1) - \theta_1 \\ \dot{\theta}(t_1) - \dot{\theta}_1 \end{bmatrix} = \mathbf{0}. \quad (B1)$$

The constraint (B1) can be augmented to the cost function by Lagrange multiplier vector $\pi = [\pi_1, \pi_2]^T$,

$$C = \pi^T N(\theta(t_1)) + \int_{t_0}^{t_f} \{L + \lambda^T (\dot{\theta} - f(\theta, u))\} dt, \quad (B2)$$

Where $\theta(t)$ is a vector function of state variables, $\theta(t) = [\theta, \omega, \gamma]^T$, u is a control variable, and $\lambda(t)$ is a co-state vector function.

The system equation $\dot{\theta} = f(\theta, u)$ is defined as follows.

$$\dot{\theta} = \omega, \dot{\omega} = \gamma, \dot{\gamma} = u. \quad (B3)$$

The cost function is expressed by

$$L = \frac{1}{2} u^2. \quad (B4)$$

The optimization problem can be solved by Pontryagin maximum principle. The optimal solution is obtained by allowing discontinuities in the co-state function.

We let t_1^- signify just before t_1 and t_1^+ signify just after t_1 . Therefore, the following conditions can be obtained.

$$\pi^T \frac{\partial N}{\partial \theta(t_1)} + \lambda^T(t_1^-) - \lambda^T(t_1^+) = 0, \quad (B5)$$

$$\dot{\lambda} = - \left(\frac{\partial f}{\partial \theta} \right)^T \lambda, \quad (B6)$$

$$\lambda^T \frac{\partial f}{\partial u} = \frac{\partial L}{\partial u}. \quad (B7)$$

From Equation (B5), we get the following condition.

$$\lambda(t_1^+) = \lambda(t_1^-) + [\pi_1, \pi_2, 0]^T. \quad (B8)$$

Different functions $\theta^-(t)$ and $\theta^+(t)$ are defined before t_1 and after t_1 .

$$\theta(t) = \begin{cases} \theta^-(t) & (t_0 \leq t \leq t_1) \\ \theta^+(t) & (t_1 \leq t \leq t_f) \end{cases}. \quad (B9)$$

$\theta(t)$ must be satisfied with following conditions,

$$\begin{aligned} \theta^+(t_1) &= \theta^-(t_1) = \theta_1, \\ \dot{\theta}^+(t_1) &= \dot{\theta}^-(t_1) = \dot{\theta}_1, \\ \ddot{\theta}^+(t_1) &= \ddot{\theta}^-(t_1), \\ \ddot{\theta}^+(t_1) &= \ddot{\theta}^-(t_1), \\ \frac{d^4}{dt^4} \theta^+(t_1) &= \frac{d^4}{dt^4} \theta^-(t_1) + \pi_2, \\ \frac{d^5}{dt^5} \theta^+(t_1) &= \frac{d^5}{dt^5} \theta^-(t_1) + \pi_1, \\ \frac{d^6}{dt^6} \theta^+(t_1) &= \frac{d^6}{dt^6} \theta^-(t_1) = 0. \end{aligned} \quad (B10)$$

Considering the condition (B10), the optimal solution can be expressed as follows,

$$\theta(t) = \begin{cases} a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 & (t_0 \leq t \leq t_1) \\ a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 & \\ + \frac{1}{24} \pi_2 (t - t_1)^4 + \frac{1}{120} \pi_1 (t - t_1)^5 & (t_1 \leq t \leq t_f) \end{cases} \quad (\text{B11})$$

9. References

- [1] T. Sugihara, Y. Nakamura and H. Inoue (2002) Real-time humanoid motion generation through ZMP manipulation based on inverted pendulum control. in IEEE Int. Conf. Robotics and Automation (ICRA). 1404–1409.
- [2] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi and H. Hirukawa (2001) The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. in IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS). 239–246.
- [3] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujikawa, K. Harada, K. Yokoi and H. Hirukawa (2003) Biped walking pattern generation by using preview control of zero-moment point. in IEEE Int. Conf. Robotics and Automation (ICRA). 1620–1626.
- [4] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujikawa, K. Harada, K. Yokoi and H. Hirukawa (2003) Resolved momentum control: humanoid motion planning based on the linear and angular momentum. in IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS). 1644–1650.
- [5] S-H. Lee, A. Goswami and B. Choi (eds.) (2009) The reaction mass pendulum (RMP) model for humanoid robot gait and balance control, Humanoid Robots. Rjeka : InTech. 167–186.
- [6] S. Kudoh, T. Komura and K. Ikeuchi (2002) The dynamic postural adjustment with the quadratic programming method. in IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS). 2563–2568.
- [7] S. Piao, Y. Liu, W. Zhao and Q. Zhong (2011) Application and Research of Humanoid Robot Based on Second-Order Cone Programming. Int. journal of Advanced Robotics Systems 8(2) :22–28.
- [8] S. Miossec, K. Yokoi, and A. Kheddar (2006) Development of a software for motion optimization of robots - Application to the kick motion of the HRP-2 robot. in IEEE Int. Conf. Robotics and Biomimetics (ROBIO). 17–20.
- [9] S. Lengagne, N. Ramdani, and P. Fraisse (2011) Planning and Fast Replanning Safe Motions for Humanoid Robots. IEEE Transactions on Robotics. 27(6): 1095–1106.
- [10] S. Lengagne, P. Mathieu, A. Kheddar, and E. Yoshida (2010) Generation of dynamic motions under continuous constraints: Efficient computation using B-Splines and Taylor polynomials. in IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS). 698–703.
- [11] E. T. Y. Lee (1982) A simplified B-spline computation routine, Computing. 29(4): 365–371.
- [12] T. Flash, and N. Hogan (1985) The coordination of arm movements an experimentally confirmed mathematical model. The Journal of Neuroscience. 5(7): 1688–1703.
- [13] Y. Uno, M. Kawato, and R. Suzuki (1989) Formation and control of optimal trajectory in human multijoint arm movement. Biological Cybernetics. 6(2): 89–101.
- [14] H. Miyamoto, S. Schaal, F. Gandolfo, H. Gomi, Y. Koike, R. Osu, E. Nakano, Y. Wada, and M. Kawato (1996) A Kendama Learning Robot Based on Bi-directional Theory. Neural Networks. 9(8): 1281–1302.
- [15] R. Hettich, and K. O. Kortanek (1993) Semi-Infinite Programming: theory, methods, and applications. SIAM Review. 35(3): 380–429.
- [16] C-H. Sung, T. Kagawa, and Y. Uno (2011) Planning of Kicking Motion with Via-Point Representation for Humanoid Robots. in IEEE Int. Conf. Ubiquitous Robots and Ambient Intelligence (URAI). 362–367.
- [17] A. Piazzzi, and A. Visioli (2000) Global minimum-jerk trajectory planning of robot manipulators. IEEE Transactions on Industrial Electronics. 47(1): 140–149.
- [18] C.G.L. Bianco, and A. Piazzzi (2002) Minimum-time trajectory planning of mechanical manipulators under dynamic constraints. International Journal of Control 75(13): 967–980.
- [19] R. M. Murray, Z. Li, and S. S. Sastry (1994) A Mathematical Introduction to Robotic Manipulation. CRC Press.
- [20] M. Vukobratovic, and B. Borovac (2004) Zero-Moment Point - thirty five years of its life. International Journal of Humanoid Robotics. 1(1): 157–173.
- [21] Y. Xu and H. Mellmann (2010) Adaptive Motion Control: Dynamic Kick for a Humanoid Robot. Advances in Artificial Intelligence, Lecture Notes in Computer Science. 6359 : 392–399.
- [22] JH. Choi, BY So, BY Yi, W. Kim and IH Suh (2005) Impact Based Trajectory Planning of a Soccer Ball in a Kicking Robot. in IEEE Int. Conf. Robotics and Automation (ICRA). 2834–2840.