2013 13th IEEE-RAS International Conference on
Humanoid Robots (Humanoids).
October 15 - 17, 2013. Atlanta, GA

Impulsive pedipulation of a spherical object for reaching a 3D goal position

Rafael Cisneros[1], Kazuhito Yokoi[2] and Eiichi Yoshida[3]

***Abstract*— The aim of this paper is to develop an algorithm that enables a humanoid robot to perform an impulsive pedipulation of a spherical object by using its foot, so that the object reaches a desired 3D goal position, taking into account some constraints imposed at the moment of the impact. This is done by planning a suitable motion of the humanoid robot that exerts the required impact conditions on this object. Then, we take the free kick in soccer as a case of study that represents one possible application of this algorithm. Finally, we provide a simulation example that intends to show its validity.**

I. Introduction

Manipulation, from the Latin *manus* "hand" plus the root *plere* "to fill", is defined as the sense of skillful handling of objects by means of the hands, or by any mechanical means.

One of the main purposes of traditional robots is the manipulation of objects in the environment by means of an *end effector*, whose shape and functionality depends highly on the task it is meant for. Many of them manipulate objects by grasping them, moving, and then releasing them. Other types of manipulation (non-grasping) may include the use of attractive forces (electromagnets, suction pads, etc.) as well as repulsive forces. Within this last category, some robots make use of pushing strategies (*non-impulsive*) to achieve the desired goal, while others make use of striking strategies (*impulsive*), which basically give an initial linear and angular velocity to the object, which then moves subject to forces and constraints [1] [2] [3].

Humanoid robots may, or may not have a proper grasping system for skillfully handling objects, as the focus of most researchers is the locomotion system. Having this into account, it is worth to consider the use of repulsive forces for non-grasping manipulation, by using its hands or feet.

This last skillful handling of the objects could even be described with the term *pedipulation*, from the Latin *pes* (genitive *pedis*) "foot", which would give an extra functionality to the locomotion system, as well as an extra challenge: to keep balance while performing both tasks in the presence new forces.

[1]R. Cisneros is with the Graduate School of Systems and Information Engineering, University of Tsukuba, 305-0006 Tsukuba, Japan and with the CNRS-AIST JRL (Joint Robotics Laboratory), UMI 3218/CRT, 305-8568 Tsukuba, Japan, rafael.cisneros at aist.go.jp

[2]K. Yokoi is with the CNRS-AIST JRL (Joint Robotics Laboratory), UMI 3218/CRT, 305-8568 Tsukuba, Japan and the Graduate School of Systems and Information Engineering, University of Tsukuba, 305-0006 Tsukuba, Japan, kazuhito.yokoi at aist.go.jp

[3]E. Yoshida is with the CNRS-AIST JRL (Joint Robotics Laboratory), UMI 3218/CRT, 305-8568 Tsukuba, Japan, e.yoshida at aist.go.jp

This term has been used previously in the robotics literature, referring to the footstep planning for a humanoid [4] and to describe a human or robot foot's action when it is used to juggle objects [5], but not in a generalized way if handling objects with the feet while maintaining balance.

It is then our interest to research on pedipulation, using non-impulsive or impulsive strategies, as it is not uncommon in our daily lives, or for actual applications of humanoid robots, as it is the case of the kicking motion in soccer: one possible application of this research.

This motion has been previously addressed in works as [6] [7] [8] [9] [10], but mainly focused in the motion itself without taking into account the way to achieve a specific goal. [11] deals with the trajectory of a ball (idealized as a particle) resulting from a kicking motion, without solving the inverse problem. This paper deals with this inverse problem in order to get a feasible kicking motion, such that the ball considered as a rigid body reaches a specific goal.

II. Problem Statement

Taking the case of soccer as a typical example of impulsive pedipulation, this work focuses on the free kick performed by a humanoid robot. The ball should reach a desired 3D goal position with a desired approaching angle and angular velocity, considering the constraints imposed by the impact's nature. In order to solve this problem we propose the algorithm shown in Fig. 1 and explained as follows:

1) Given the 3D *goal position* of the ball, its *approaching angle* and *final angular velocity*, it is first necessary to calculate the required *initial linear and angular velocity* of the ball at its home position.

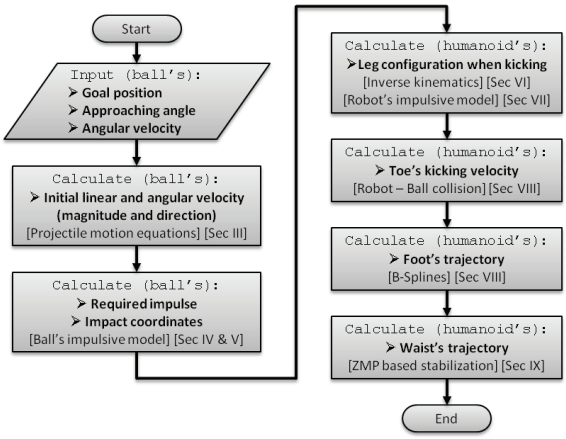

Fig. 1. Impulsive pedipulation algorithm

2) Assuming that the ball is originally in a stationary state, it is necessary to calculate the *required impulse* needed to produce the momentum required to perform the desired motion, as well as where to apply it; that is, the *impact coordinates*.
3) Knowing where to apply the required impulse, the required *configuration of the leg* of the robot at the moment of impact can be calculated, as well as the required *kicking velocity* of the foot.
4) Finally, a suitable *trajectory of the foot* should be chosen, so that it can exert the desired kick motion, while maintaining the robot's stability by means of a proper *trajectory of the waist*.

The next sections deal with every one of these stages.

III. Projectile Motion

Let us define two frames: the world frame $\{W\}$ where the Z axis points up, and the ball frame $\{B\}$ which is rigidly attached to the ball. Any point of the ball that is described in $\{B\}$ will be denoted by the corresponding position vector and a leading superscript indicating that it is referenced to it [12]. Vectors that are not denoted by any leading superscript should be considered as being described in $\{W\}$.

Let us assume that the ball will follow a projectile motion (under the influence of gravity only); that is, the effect of various aero dynamical forces, as the drag force and the "Magnus effect" [11], will be neglected given the low impulse that the actual humanoid robots can exert on the ball.

Given a 3D goal position of the ball $\boldsymbol{P_G}$, we can use the standard projectile motion equations that describe the trajectory of the ball's position (its center) $\boldsymbol{P_B}(t)$ to find a suitable initial velocity vector $\boldsymbol{v_0}$.

This trajectory is given by

$$\boldsymbol{P_B}(t) = \boldsymbol{P_B}(t_0) + \boldsymbol{v_0}t - \frac{1}{2}\boldsymbol{g}t^2 \tag{1}$$

Where t_0 is the time of the impact, $\boldsymbol{g} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T$ is the gravity vector and g is the gravity's acceleration, whereas the initial linear velocity vector $\boldsymbol{v_0}$ may be described by its magnitude v_0 and its direction $\{\phi_m, \theta_{m0}\}$ (Fig. 2): ϕ_m is defined as the angle between the trajectory's projection on the ground and the X axis of $\{W\}$ and θ_{m0} as the *angle of launch*. Such that,

$$\boldsymbol{v_0} = \begin{bmatrix} v_{0x} \\ v_{0y} \\ v_{0z} \end{bmatrix} = \begin{bmatrix} v_0 \cos\phi_m \cos\theta_{m0} \\ v_0 \sin\phi_m \cos\theta_{m0} \\ v_0 \sin\theta_{m0} \end{bmatrix} \tag{2}$$

Then, considering that T is the time needed to attain the goal, we get $\boldsymbol{P_G} = \begin{bmatrix} P_{Gx} & P_{Gy} & P_{Gz} \end{bmatrix}^T$ by evaluating $\boldsymbol{P_G} = \boldsymbol{P_B}(T)$.

The angle ϕ_m gives us the orientation of the *plane of motion* (where the motion occurs) with respect to the XZ plane, so that this one is constant during the whole trajectory and can be directly calculated from the goal position as

$$\phi_m = \arctan 2\left(P_{Gy}, P_{Gx}\right) \tag{3}$$

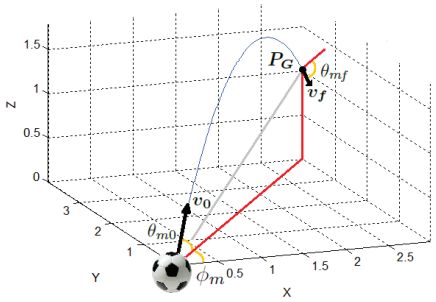

Fig. 2. Projectile motion

However, the other three variables (T, v_0, θ_{m0}) cannot be directly calculated because the three equations are not independent.

There is an infinite set of possible trajectories capable of accomplishing the required goal, unless we specify another constraint for the problem. We select the *approaching angle* θ_{mF} (Fig. 2) as a constraint, which can be defined as the direction of the ball's movement at the goal position and within the plane of motion; that is, the angle between the velocity vector at that point, $\boldsymbol{v_f}$, and the XY plane of $\{W\}$.

The velocity of the ball at the goal position is obtained by $\boldsymbol{v_f} = \frac{d}{dt}\boldsymbol{P_B}(T) = \begin{bmatrix} v_{fx} & v_{fy} & v_{fz} \end{bmatrix}^T$, such that

$$\tan\theta_{mf} = \frac{v_{fz}}{\sqrt{v_{fx}^2 + v_{fy}^2}} = \frac{v_0 \sin\theta_{m0} - gT}{v_0 \cos\theta_{m0}} \tag{4}$$

Let us refer to (1) after substituting (2) and evaluating it with T to get $\boldsymbol{P_G}$, as well as (4). We have a system of equations that can be solved for the required angle of launch θ_{m0} and the magnitude of the initial linear velocity v_0:

$$\theta_{m0} = \arctan\left(2P_{Gz}\frac{\cos\phi_m}{P_{Gx}} - \tan\theta_{mf}\right) \tag{5}$$

$$v_0 = \sqrt{\frac{P_{Gx}}{\cos\phi_m}\frac{g\left(1+\tan^2\theta_{m0}\right)}{\tan\theta_{m0} - \tan\theta_{mf}}} \tag{6}$$

And because there are no aero dynamical forces considered that may change the angular velocity of the ball during the trajectory, the initial angular velocity ω_0 is equal to the final angular velocity ω_f at the goal position; that is,

$$\boldsymbol{\omega_0} = \boldsymbol{\omega_f} \tag{7}$$

IV. Ball's Impulsive Contact Model

Having obtained the initial linear and angular velocity vectors, and assuming that the ball is originally steady, it is necessary to calculate the *impulse* $\hat{\boldsymbol{f}}_{\boldsymbol{P}}$ needed to produce the momentum required to initiate the ball's desired motion, as well as the point on the ball to apply that impulse; that is, the *impact coordinates* (ϕ_0, θ_0), as shown in Fig. 3.

This is done by formulating and solving the dynamical model of the ball at the moment of the impact; that is, considering the external impulse.

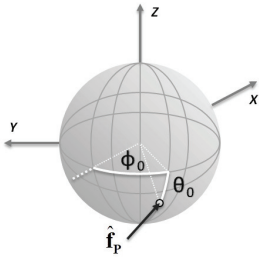

Fig. 3. Definition of the impact coordinates in $\{W\}$

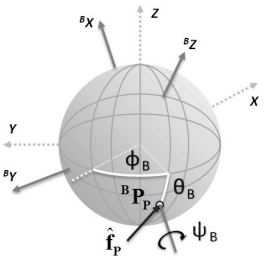

Fig. 4. Relationship between the ball frame and the world frame

A. Orientation of the ball

Idealizing the ball as a plain sphere, it doesn't have any inherent orientation. However, in order to introduce the impact coordinates (ϕ_0, θ_0) into the model, we can consider that the frame $\{B\}$ is oriented such that the ball is always impacted at the point ${}^{B}\boldsymbol{P_P}$, described in that frame; that is,

$$ {}^{B}\boldsymbol{P_P} = \begin{bmatrix} -r & 0 & 0 \end{bmatrix}^T \tag{8}$$

Then, if we describe the orientation of $\{B\}$ with respect to $\{W\}$ by using the Euler ZYX convention and the orientation coordinates $(\phi_B, \theta_B, \psi_B)$ (Fig. 4), we will have the following correspondence with the impact coordinates:

$$ \phi_0 = \phi_B(t_0) \qquad \theta_0 = \theta_B(t_0) \tag{9}$$

The rotation matrix that describes the orientation of $\{B\}$ with respect to $\{W\}$ is denoted by ${}^{W}_{B}\boldsymbol{R}(\phi_B, \theta_B, \psi_B)$, according to the notation introduced in [12].

Let us remark that only when $\theta_B = \pm\frac{\pi}{2}$ the matrix ${}^{W}_{B}\boldsymbol{R}$ becomes singular; that is, the highest and lowest points on the ball, which are not of interest for the task.

B. Lagrange equation of the ball

Let us describe the configuration of the ball $\boldsymbol{q_B} \in \mathbb{R}^6$ by using the set of *generalized coordinates* given by the position of its center, $\boldsymbol{P_B}$, and its orientation coordinates:

$$ \boldsymbol{q_B} = \begin{bmatrix} P_{Bx} & P_{By} & P_{Bz} & \phi_B & \theta_B & \psi_B \end{bmatrix}^T \tag{10}$$

If the mass of the ball is denoted as m_B and its Inertia Tensor with respect to its center of gravity as $\boldsymbol{I_B}$, then its kinetic energy E_{KB} is given by:

$$ E_{KB} = \frac{1}{2} m_B v_B^2 + \frac{1}{2} \boldsymbol{\omega_B}^T \boldsymbol{I_B} \boldsymbol{\omega_B} \tag{11}$$

Where v_B is the magnitude of the linear velocity of the center of gravity of the ball and $\boldsymbol{\omega_B}$ is its angular velocity.

This angular velocity is related to the rate of change of the orientation coordinates, $\left(\dot{\phi}_B, \dot{\theta}_B, \dot{\psi}_B\right)$, given that the *skew-symmetric matrix* of the angular velocity of the ball, denoted as $\hat{\boldsymbol{\omega}}_{\boldsymbol{B}}$ (in order to differentiate it from the vector $\boldsymbol{\omega_B}$), is a function of ${}^{W}_{B}\dot{\boldsymbol{R}}$:

$$ \hat{\boldsymbol{\omega}}_{\boldsymbol{B}} = \left({}^{W}_{B}\dot{\boldsymbol{R}}\right)\left({}^{W}_{B}\boldsymbol{R}\right)^T \tag{12}$$

Finally, the Inertia Tensor of the ball of radius r can be expressed as $\boldsymbol{I_B} = \rho m_B r^2 \boldsymbol{I_3}$, where $\boldsymbol{I_3} \in \mathbb{R}^3$ is an identity matrix and ρ is defined as the construction coefficient of the ball. The potential energy is given by

$$ E_{PB} = m_B \boldsymbol{g}^{\boldsymbol{T}} \boldsymbol{P_B} \tag{13}$$

Having done this, the Lagrangian can be calculated as $L_B = E_{KB} - E_{PB}$, so that the dynamical equation of the ball at the moment of the impact can be expressed as:

$$ \frac{d}{dt}\frac{\partial L_B}{\partial \dot{\boldsymbol{q}}_{\boldsymbol{B}}} - \frac{\partial L_B}{\boldsymbol{q_B}} = \boldsymbol{Q_B} \tag{14}$$

C. Description of the impulsive forces

$\boldsymbol{Q_B}$ represents the *generalized impulsive force* created by the impact and, in order to calculate it, we need first to express ${}^{B}\boldsymbol{P_P}$ (defined in (8)) in $\{W\}$:

$$ \boldsymbol{P_P} = \boldsymbol{P_B} + {}^{W}_{B}\boldsymbol{R}\,{}^{B}\boldsymbol{P_P} \tag{15}$$

From which the Jacobian $\boldsymbol{J_{BP}}(\boldsymbol{q_B}(t))$ can be calculated by differentiating the last expression with respect to t.

The transpose of this Jacobian, $\boldsymbol{J^T_{BP}}$, relates the force applied to the point $\boldsymbol{P_P}$, $\boldsymbol{f_P}$, with $\boldsymbol{Q_B}$:

$$ \boldsymbol{Q_B} = \boldsymbol{J^T_{BP}} \boldsymbol{f_P} \tag{16}$$

D. External impulse modeling

One way of modeling the external impulse is suggested in [13]. The ball's dynamical equation can be expressed as

$$ \boldsymbol{M_B}\ddot{\boldsymbol{q}}_{\boldsymbol{B}}(t) + \boldsymbol{C_B} + \boldsymbol{G_B} = \boldsymbol{J^T_{BP}} \boldsymbol{f_P} \tag{17}$$

Where $\boldsymbol{M_B}(\boldsymbol{q_B}(t)) \in \mathbb{R}^{6\times 6}$ corresponds to the mass matrix, $\boldsymbol{C_B}(\boldsymbol{q_B}(t), \dot{\boldsymbol{q}}_{\boldsymbol{B}}(t)) \in \mathbb{R}^6$ to the coupling vector and $\boldsymbol{G_B}(\boldsymbol{q_B}(t)) \in \mathbb{R}^6$ to the gravitational one.

This dynamical model is valid only during the short time that the collision lasts; that is, from t_0 to $t_0 + \Delta t$. Let us integrate (17) in this period of time and assume that Δt is so small that it is possible to idealize it as $\Delta t \to 0$.

Under this assumption, the position and orientation of the ball remains constant during the time that the impact lasts. Also, as the velocities and the gravitational effect are assumed to be finite, the integral terms $\int_{t_0}^{t_0+\Delta t} \boldsymbol{C_B} d\tau$ and $\int_{t_0}^{t_0+\Delta t} \boldsymbol{G_B} d\tau$ become zero as $\Delta t \to 0$ [11] [13]. The integral term $\int_{t_0}^{t_0+\Delta t} \boldsymbol{f_P} d\tau$ produces a finite impulse, denoted by $\hat{\boldsymbol{f}}_{\boldsymbol{P}}$. Then we have,

$$ \boldsymbol{\Delta \dot{q}_B} = \boldsymbol{M_B^{-1}} \boldsymbol{J^T_{BP}} \hat{\boldsymbol{f}}_{\boldsymbol{P}} \tag{18}$$

Where $\boldsymbol{\Delta \dot{q}_B} = \dot{\boldsymbol{q}}_{\boldsymbol{B}}(t_0^+) - \dot{\boldsymbol{q}}_{\boldsymbol{B}}(t_0^-)$ (as $\Delta t \to 0$, t_0^- and t_0^+ will be used to refer to the time t_0 just before and after the impact, respectively). That is, it relates the impulse exerted on the ball with its linear and angular velocity just before and after the impact.

V. Required Impact Conditions

Let us consider (18). This equation is a function of $\hat{\boldsymbol{f}}_{\boldsymbol{P}}$, ϕ_B, θ_B, ψ_B and $\dot{\boldsymbol{q}}_{\boldsymbol{B}}\left(t_0^+\right)$ (as the ball is steady just before the impact, $\boldsymbol{\Delta}\dot{\boldsymbol{q}}_{\boldsymbol{B}} = \dot{\boldsymbol{q}}_{\boldsymbol{B}}\left(t_0^+\right)$), where $\dot{\boldsymbol{q}}_{\boldsymbol{B}}\left(t_0^+\right)$ is a function of the initial linear velocity $\boldsymbol{v_0}$ and the initial angular velocity $\boldsymbol{\omega_0}$ (which depends on $\dot{\phi}_B\left(t_0^+\right)$, $\dot{\theta}_B\left(t_0^+\right)$, $\dot{\psi}_B\left(t_0^+\right)$).

By expanding (18) we get six equations corresponding to each of its components: the first three relate the applied impulse to the initial linear momentum of the ball, such that we can directly calculate $\hat{\boldsymbol{f}}_{\boldsymbol{P}}$ as

$$\hat{\boldsymbol{f}}_{\boldsymbol{P}} = m_B \boldsymbol{v_0} \tag{19}$$

On the other hand, the following three equations give us the relationship for its initial angular momentum, from where it is possible to calculate the necessary impact coordinates $\phi_0 = \phi_B\left(t_0^+\right)$ and $\theta_0 = \theta_B\left(t_0^+\right)$ by solving:

$$\begin{aligned} &\hat{f}_{Px} s_\phi c_\theta - \hat{f}_{Py} c_\phi c_\theta + \rho m_B r \dot{\psi}_{B0} s_\theta = \rho m_B r \dot{\phi}_{B0} \\ &\hat{f}_{Px} c_\phi s_\theta + \hat{f}_{Py} s_\phi s_\theta + \hat{f}_{Pz} c_\theta = \rho m_B r \dot{\theta}_{B0} \\ &\dot{\phi}_{B0} s_\theta = \dot{\psi}_{B0} \end{aligned} \tag{20}$$

$c_\phi = \cos\phi_{B0}$, $s_\phi = \sin\phi_{B0}$, $c_\theta = \cos\theta_{B0}$, $s_\theta = \sin\theta_{B0}$.

This can be done by means of a multiobjective optimization procedure as it is the *goal attain* method [14].

A. Main components of the impulse

The impulse $\hat{\boldsymbol{f}}_{\boldsymbol{P}}$ has two main components (Fig. 5): (i) a *normal component* ($\hat{f}_{Pn}$) which is directly related to the change in relative motion (in the normal direction) of the *points* that come into contact, and (ii) a *tangential component* ($\hat{f}_{Pt}$) which has to be produced just by the effect of friction [15] [16]; such that,

$$\hat{\boldsymbol{f}}_{\boldsymbol{P}} = \hat{f}_{Pn}\hat{\boldsymbol{n}} + \hat{f}_{Pt}\hat{\boldsymbol{t}} \tag{21}$$

This means that there is a constraint that should be taken into account when solving (20): The magnitude of the tangential component of the impulse is constrained by the *coefficient of static friction*, μ_s; such that we need to consider the following inequality:

$$\hat{f}_{Pt} \leq \mu_s \hat{f}_{Pn} \tag{22}$$

The vector $\hat{\boldsymbol{n}}$ can be calculated from (15) as

$$\hat{\boldsymbol{n}} = \frac{\boldsymbol{P_B} - \boldsymbol{P_P}}{\|\boldsymbol{P_B} - \boldsymbol{P_P}\|} = \begin{bmatrix} c_\phi c_\theta \\ s_\phi c_\theta \\ -s_\theta \end{bmatrix} \tag{23}$$

While the vector $\hat{\boldsymbol{t}}$ can then be calculated from (21) by knowing $\boldsymbol{f_P}$ and $\hat{\boldsymbol{n}}$.

B. The zero initial angular velocity case

As we are not considering any aero dynamical effect that depends on the angular motion, it is possible to simplify (20) by considering a desired zero initial angular velocity, $\boldsymbol{\omega_0} = \boldsymbol{0}$; that is, $\dot{\phi}_B(t_0) = \dot{\theta}_B(t_0) = \dot{\psi}_B(t_0) = 0$, so that we have

$$\phi_{B0} = \arctan 2\left(v_{0y}, v_{0x}\right) \tag{24}$$

$$\theta_{B0} = -\arctan 2\left(v_{0z}, \sqrt{v_{0x}^2 + v_{0y}^2}\right) \tag{25}$$

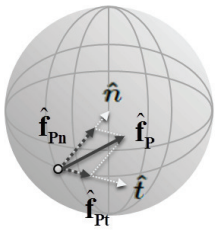

Fig. 5. Normal and tangential components of the impulse

VI. Humanoid's Configuration

Let us consider a humanoid robot which will perform the kicking motion to exert on the ball the desired impact conditions. Also, let us consider three frames: $\{S\}$, $\{C\}$ and $\{K\}$, rigidly attached to the support foot, the waist and the kicking foot, respectively.

By knowing the initial position of the ball, $\boldsymbol{P_B}\left(t_0^-\right)$ and the impact coordinates (ϕ_0, θ_0) we can calculate the impact point $\boldsymbol{P_P}$ (in the world frame) as

$$\boldsymbol{P_P} = \boldsymbol{P_B}\left(t_0^-\right) - r\hat{\boldsymbol{n}} \tag{26}$$

Then, proposing for the humanoid: (i) the position and orientation of the support foot, $\left(\boldsymbol{P_S}, {}^{\boldsymbol{W}}_{\boldsymbol{S}}\boldsymbol{R}\right)$, (ii) the position and orientation of the waist at the moment of the impact, $\left(\boldsymbol{P_C}, {}^{\boldsymbol{W}}_{\boldsymbol{C}}\boldsymbol{R}\right)$, (iii) the point on the kicking foot that will impact the ball, referred from now on as the *"toe"* and described in $\{K\}$, ${}^{\boldsymbol{K}}\boldsymbol{P_P}$ (Fig. 6), and (iv) the orientation of the kicking foot in that moment, ${}^{\boldsymbol{W}}_{\boldsymbol{K}}\boldsymbol{R}$, we can calculate the configuration of the robot $\boldsymbol{q_R}$ (the joint values) at the moment of the impact, by solving its *inverse kinematics* problem.

VII. Humanoid's Impulsive Contact Model

The dynamic model of the humanoid robot can be expressed in a similar way as for the ball, (17), as

$$\boldsymbol{M_R}\ddot{\boldsymbol{q}}_{\boldsymbol{R}}(t) + \boldsymbol{C_R} + \boldsymbol{G_R} = \boldsymbol{\tau_R} - \boldsymbol{J}_{\boldsymbol{RP}}^{\boldsymbol{T}}\boldsymbol{f_P} \tag{27}$$

Where $\boldsymbol{q_R}(t)$ contains the joint values and $\boldsymbol{\tau_R}(t)$, the corresponding torques. By the third law of Newton, $\boldsymbol{f_P}$ should have the same magnitude and opposite direction, as indicated with the minus sign.

Integrating (27) from t_0 to $t_0 + \Delta t$ and assuming $\Delta t \to 0$,

$$\boldsymbol{\Delta}\dot{\boldsymbol{q}}_{\boldsymbol{R}} = -\boldsymbol{M}_{\boldsymbol{R}}^{-\boldsymbol{1}}\boldsymbol{J}_{\boldsymbol{RP}}^{\boldsymbol{T}}\hat{\boldsymbol{f}}_{\boldsymbol{P}} \tag{28}$$

This expression is similar to the one obtained for the ball.

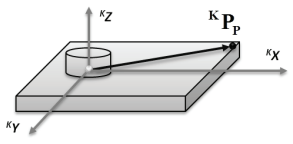

Fig. 6. Description of the "toe" in the kicking foot frame

VIII. Robot-Ball Collision Model

Knowing where to apply the impulse, as well as the configuration of the ball and the robot $(\boldsymbol{q_B}, \boldsymbol{q_R})$ at the moment of impact, we are able to calculate the required approaching velocity of the "toe" (the point of contact of the foot with the ball), which will exert the necessary impulse that will start the ball's motion.

We will assume that the bodies in contact at the moment of collision are rigid. Then the mathematical theory of rigid body collisions may be used [13].

When two rigid bodies collide, the relationship between the velocity of both *at the contact point* ($\boldsymbol{P_P}$) [16] just before and after the impact can be expressed by means of the following expression [11]:

$$\left(\boldsymbol{\Delta v_{RP}} - \boldsymbol{\Delta v_{BP}}\right)^T \hat{\boldsymbol{n}} = -\left(1+e\right)\left(\boldsymbol{v_{RP}} - \boldsymbol{v_{BP}}\right)^T \hat{\boldsymbol{n}} \tag{29}$$

Where e is the Coefficient of Restitution (CoR), $\boldsymbol{v_{BP}}$ and $\boldsymbol{v_{RP}}$ stand for the velocity of the point $\boldsymbol{P_P}$ on the ball and on the robot's toe, respectively, along the unitary vector $\hat{\boldsymbol{n}}$ that is normal to the contact surface.

The vectors $\boldsymbol{\Delta v_{BP}}$ and $\boldsymbol{\Delta v_{RP}}$ are calculated with the aid of (18) and (28), respectively, and the corresponding Jacobian:

$$\boldsymbol{\Delta v_{BP}} = \boldsymbol{J_{BP}} \boldsymbol{\Delta \dot{q}_B} = \boldsymbol{J_{BP}} \boldsymbol{M_B^{-1}} \boldsymbol{J_{BP}^T} \hat{\boldsymbol{f}}_{\boldsymbol{P}} \tag{30}$$

$$\boldsymbol{\Delta v_{RP}} = \boldsymbol{J_{RP}} \boldsymbol{\Delta \dot{q}_R} = -\boldsymbol{J_{RP}} \boldsymbol{M_R^{-1}} \boldsymbol{J_{RP}^T} \hat{\boldsymbol{f}}_{\boldsymbol{P}} \tag{31}$$

Where $\boldsymbol{M_R}\left(\boldsymbol{q_R}\left(t_0^-\right)\right)$ and $\boldsymbol{M_B}\left(\boldsymbol{q_B}\left(t_0^-\right)\right)$ are the mass matrices of the robot and the ball, and $\boldsymbol{J_{RP}}\left(\boldsymbol{q_R}\left(t_0^-\right)\right)$ and $\boldsymbol{J_{BP}}\left(\boldsymbol{q_B}\left(t_0^-\right)\right)$ are the corresponding Jacobians that relate the velocity of the point $\boldsymbol{P_P}$ (in each object) with the rate of change of the generalized coordinates. Notice that these matrices are a function of $\boldsymbol{q_R}$ and $\boldsymbol{q_B}$ at the moment of the impact, which were previously computed.

Substituting these expressions into (29) and assuming that the ball is originally at rest we can calculate the *normal* component of $\boldsymbol{v_{RP}}$, the approaching velocity of the toe:

$$\boldsymbol{v_{RP}^T} \hat{\boldsymbol{n}} = \frac{\hat{f}_{Pn}}{1+e} \hat{\boldsymbol{n}}^T \boldsymbol{N} \hat{\boldsymbol{n}} + \frac{\hat{f}_{Pt}}{1+e} \hat{\boldsymbol{n}}^T \boldsymbol{N} \hat{\boldsymbol{t}} \tag{32}$$

Where $\boldsymbol{N_R} = \boldsymbol{J_{RP}} \boldsymbol{M_R^{-1}} \boldsymbol{J_{RP}^T}$, $\boldsymbol{N_B} = \boldsymbol{J_{BP}} \boldsymbol{M_B^{-1}} \boldsymbol{J_{BP}^T}$ and $\boldsymbol{N} = \boldsymbol{N_R} + \boldsymbol{N_B}$.

In order to calculate the tangential component of this velocity it is necessary to consider that the required frictional impulse is less than required to begin sliding, as (22) holds, so that the foot sticks to the ball and there will not be any relative motion between them over the infinitesimal time that the impact lasts. Then, the foot has to follow the same tangential velocity of the point of the ball $\boldsymbol{P_P}$ when the last one starts its movement, such that

$$\boldsymbol{v_{RP}^T} \hat{\boldsymbol{t}} = \boldsymbol{\Delta v_{BP}^T} \hat{\boldsymbol{t}} = \hat{f}_{Pn} \hat{\boldsymbol{n}}^T \boldsymbol{N_B} \hat{\boldsymbol{t}} + \hat{f}_{Pt} \hat{\boldsymbol{t}}^T \boldsymbol{N_B} \hat{\boldsymbol{t}} \tag{33}$$

In this way we can calculate the required velocity of the point $\boldsymbol{P_P}$ on the foot at the moment of impact with the ball, so that it acquires the desired motion.

IX. Kicking Motion Planning

Once the required velocity of the toe is known, we need to generate a suitable 3D trajectory that satisfies the position and velocity of the toe at the moment of the impact.

This can be accomplished by shaping this trajectory with *cubic B-Spline curves* as these ones have very nice properties: (i) the curve is entirely contained in the convex hull of its control polyline, and (ii) the curve can be locally controlled [17].

A B-Spline is an approximating curve. Its shape is determined by its control points but the curve itself does not pass through those. However, it is possible to interpolate a set of data points $\boldsymbol{K_0}, \ldots, \boldsymbol{K_n}$ with a curve made up of n segments, which starts and ends with given velocities $\boldsymbol{v_0}$ and $\boldsymbol{v_n}$, and is travelled during a time τ, by solving a linear system of equations, as shown in [18].

A. Trajectory planning

By using B-Splines we can now construct a trajectory for the kicking foot. Although there are many ways to achieve this motion, generally it may be divided into two phases, each one described by a different B-Spline: (i) the *kicking phase*, which detaches the foot from the ground (with an initial zero velocity) and drives it to the kicking position $\boldsymbol{P_P}$ with the required velocity $\boldsymbol{v_{RP}}$, and (ii) the *recovering phase*, which drives the foot moving with a velocity $\boldsymbol{v_{RP}}$ to the landing position (with a zero velocity).

It is worth mentioning that sometimes the trajectory may not be contained inside of the leg's workspace, unless we vary the time span τ of each curve.

In general, a shorter τ will lead to shorter curves, such that we can fit them inside the workspace. However, shorter values for τ imply that the foot will reach the desired velocity in a shorter time; that is, they will require higher acceleration values, which may be physically impossible to a achieve by using a real robot.

B. Stabilization

Once the trajectory of the foot is decided and the leg motion is produced, the robot should maintain its stability during its motion. This can be done by independently moving the waist of the robot (or its upper limbs) such that the ZMP remains inside the polygon of support [19].

In this work, we used the method proposed in [20] and implemented as in [21] to generate a waist trajectory capable of stabilizing the robot's motion.

This method, however, also modifies the waist position during the impact, which in turns changes the impulse exerted ($\boldsymbol{v_{RP}}$ is a function of $\boldsymbol{M_R}$ and $\boldsymbol{J_{RP}}$, as seen in (31)). The whole-body balance can be taken into account through generalized IK, for example by using the COG Jacobian [22]. However, in this paper we adopted a simple method that iteratively runs the algorithm just described, updating the waist position suggested by the stabilizing process, until this position remains unchanged during the impact.

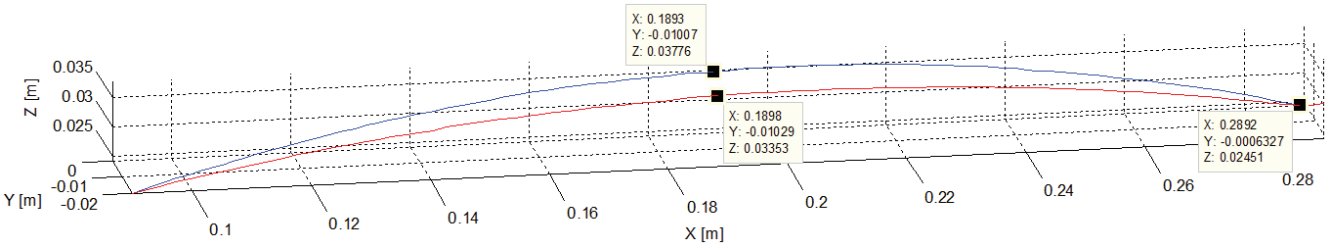

Fig. 7. Desired ball's Trajectory (blue) vs. obtained one (red)

X. Simulation Results

In order to test the developed algorithm we simulated a kicking motion, performed by the small-sized commercial humanoid robot GR-001. This robot is 0.255 m tall and weights about 0.9 kg. It has 20 dof: six at each leg, three at each arm, one at the waist and one at the neck. The legs are driven by servos Futaba RS303MR (6.5 kgf.cm, 0.11 sec/60°) while the rest is driven by servos Futaba RS304MD (5.0 kgf.cm, 0.16 sec/60°).

Given the size and weight of the robot, a rubber ball with a radius $r = 0.0243$ m and a mass $m_B = 0.010$ kg was chosen. The ball's construction coefficient is taken as $\rho = 2/5$ and the coefficient of static friction is arbitrarily considered as $\mu_s = 0.5$. The CoR of the ball was measured using the procedure explained in [23], such that $e = 0.74$.

Considering that the robot will perform a kick by using his right leg, the position of support $\boldsymbol{P_S}$ and the position of the waist during the impact $\boldsymbol{P_C}$ are proposed as:

$$\boldsymbol{P_S} = \begin{bmatrix} 0 & 0.0196 & 0 \end{bmatrix}^T \text{m} \quad (34)$$

$$\boldsymbol{P_C} = \begin{bmatrix} -0.0282 & 0.0258 & 0.145 \end{bmatrix}^T \text{m} \quad (35)$$

Initially, the position of the kicking toe is given by $\boldsymbol{P_{RP0}}$ whereas for the ball it is $\boldsymbol{P_{B0}}$. These ones are set up to be:

$$\boldsymbol{P_{RP0}} = \begin{bmatrix} 0.03048 & -0.0196 & 0 \end{bmatrix}^T \text{m} \quad (36)$$

$$\boldsymbol{P_{B0}} = \begin{bmatrix} 0.09 & -0.02 & 0.0243 \end{bmatrix}^T \text{m} \quad (37)$$

It is desired that the ball reaches the goal position:

$$\boldsymbol{P_G} = \begin{bmatrix} 0.29 & 0 & 0.0243 \end{bmatrix}^T \text{m} \quad (38)$$

That is, the ball should reach the floor 0.20 m ahead and 0.02 m to the left of its original position. Also, it is required a final angular velocity $\boldsymbol{\omega_f} = \boldsymbol{0}$ rad/s and an approaching angle of $\theta_{mf} = -0.2618$ rad.

In order to do that the ball should be given an initial linear velocity $\boldsymbol{v_0}$, which is produced by applying the impulse $\hat{\boldsymbol{f}}_{\boldsymbol{P}}$ at the point of the ball with impact coordinates $\phi_0 = 0.0997$ rad and $\theta_0 = -0.2618$ rad. These ones are calculated as:

$$\boldsymbol{v_0} = \begin{bmatrix} 1.9087 & 0.19087 & 0.51397 \end{bmatrix}^T \text{m/s} \quad (39)$$

$$\hat{\boldsymbol{f}}_{\boldsymbol{P}} = \begin{bmatrix} 0.019087 & 0.0019087 & 0.0051397 \end{bmatrix}^T \text{N} \cdot \text{s} \quad (40)$$

The toe's kicking velocity necessary to exert that impulse is calculated as

$$\boldsymbol{v_{RP}} = \begin{bmatrix} 1.3898 & 0.13898 & 0.37425 \end{bmatrix}^T \text{m/s} \quad (41)$$

The kicking motion was simulated on OpenHRP3 [24] [25], given that this one was previously augmented by us to handle realistic simulations of the ball dynamics [26]. Within this simulator, the planned motion of the robot was used as a reference by a simulated proportional derivative (PD) controller, whose gains are 8000 and 500, respectively.

Three frames of the simulation are shown in Fig. 8, where it can be seen that the ball reaches the goal position, tightly delimited by a pair of goalposts and a crossbar.

The trajectory followed by the ball is shown in Fig. 7, compared to the desired one. As we can see, the goal which is originally 0.201 m away from the start is reached with an excellent precision: the distance between the desired goal and the obtained one is 0.001043 m (representing a 0.52% of error). However, there is some discrepancy among the height attained by both trajectories, as the maximum height that the ball acquires is 88.8% of the desired one. This is because there is some positioning and following error that the PD controller is not able to suppress.

Let us look at the trajectory followed by the toe, which is shown in Fig. 9 compared to the desired one. At first glance, it seems that even when there is some positioning error the ball is hit in the right place. However, if we make a zoom in to the previous trajectory at the impact point (Fig. 10) we can observe that there is a little error, as the ball is actually hit at a slightly higher point. This is the main reason why the ball follows a lower trajectory.

Also, if we analyze the simulated kicking velocity, it is measured to be

$$\boldsymbol{v_{RP,sim}} = \begin{bmatrix} 1.3715 & 0.1359 & 0.3781 \end{bmatrix}^T \text{m/s} \quad (42)$$

Which differs from the desired one shown in (41) by an error in magnitude of about 1.17%.

XI. Conclusions and Future Work

The developed algorithm solves the inverse problem for the impact based trajectory of a spherical object satisfying its conditions needed to plan its pedipulation.

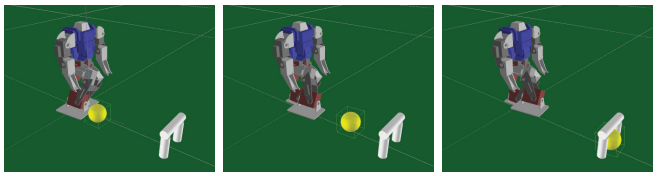

Fig. 8. Simulated kicking motion

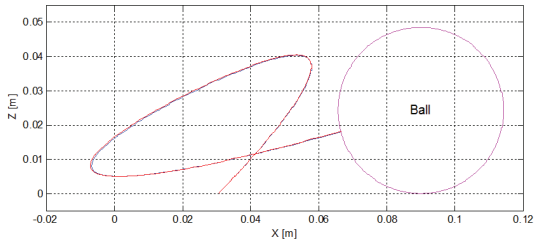

Fig. 9. Desired toe's Trajectory (blue) vs. obtained one (red) (XZ plane)

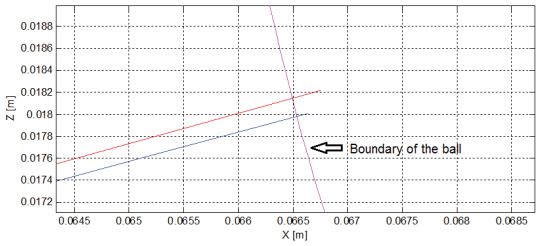

Fig. 10. Toe's Trajectory Zoom In (XZ plane)

By using this method it is possible to have control over the whole trajectory of the object, as long as it is physically possible to be attained.

Referring to the simulation, we can see that the impact conditions are actually very sensitive about the impact coordinates and velocity, so that a precise control is needed to perform the requested task with enough accuracy.

Another important point to remark is that, even when in simulation the robot can try to follow a trajectory, in reality a robot is limited by physical constraints, as there are the torque and maximum velocity of the actuators, such that many demanding tasks cannot be implemented in reality given the actual hardware.

One of those demanding tasks is to lift the ball over a considerable distance. Even when this distance is small, as in the example shown in this work, the required joint velocities overpass the 9.52 rad/sec limitation of the servos of this robot (three of them overpass 15 rad/sec). Otherwise, the trajectories would be beyond the leg's workspace.

Future work includes the application of the proposed method to a human-sized humanoid robot. The small-sized robot used in this work has large feet. Then, it is relatively easy to stabilize its motion, even by using demanding toe's trajectories. However, for a human-sized robot it is worth to include a complete study of the balance of the robot, which makes it possible to stabilize it even in the presence of disturbances as it is the impact.

Also, until now, this algorithm requires the specification of some parameters besides the goal conditions of the ball. We would like to automatically tune in those parameters by using an optimization process and, in this way, let the algorithm choose efficient ones to accomplish the desired task without the need of specifying them.

REFERENCES

[1] W. Huang and Matthew T. Mason. Experiments in impulsive manipulation. In *IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 1077–1082, May 1998.

[2] C. Zhu, Y. Aiyama, and T. Arai. Releasing manipulation with learning control. In *IEEE Int. Conf. on Robotics and Automation*, 1999.

[3] C. Zhu, Y. Aiyama, T. Arai, and A. Kawamura. Motion characteristics in releasing manipulation. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2000.

[4] T. Sugihara and Y. Nakamura. Boundary condition relaxation method for stepwise pedipulation planning of biped robots. *IEEE Trans. on Robotics*, 2009.

[5] M. Mason and K. Lynch. Dynamic manipulation. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*.

[6] J. Müler, T. Laue, and T. Röfer. Kicking a ball - modeling complex dynamic motions for humanoid robots. 2010.

[7] Felix Wenk and Thomas Röfer. Online generated kick motions for the nao balanced using inverse dynamics. In *RoboCup 2013*, 2013.

[8] S. Miossec, K. Yokoi, and A. Kheddar. Development of a software for motion optimization of robots - applicatio to kick motion of the hrp-2 robot. pages 299–304, 2006.

[9] S. Lengagne, P. Fraisse, and N. Ramdani. Planning and fast replanning of safe motions for humanoid robots: Application to a kicking motion. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 441–446, 2009.

[10] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirokawa. Resolved momentum control: Humanoid motion planning based on the linear and angular momentum. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1644–1650, 2003.

[11] J.Y. Choi, B.R. So, B.J. Yi, W. Kim, and I.H. Suh. Impact based trajectory planning of a soccer ball in a kicking robot. In *IEEE Int. Conf. on Robotics and Automation*, 2005.

[12] J.J. Craig. *Introduction to Robotics: Mechanics and Control*. Addison-Wesley, 1986.

[13] I.D. Walker. Impact configurations and measures for kinematically redundant and multiple armed robot systems. *IEEE Transactions on Robotics and Automation*, 10(5):670–683, October 1994.

[14] F. W. Gembicki. *Vector Optimization for Control with Performance and Parameter Sensitivity Indices*. PhD thesis, Case Western Reserve University, 1974.

[15] C. White. *Projectile Dynamics in Sports: Principles and Applications*. Routledge, 2011.

[16] W.J. Stronge. *Impact Mechanics*. Cambridge University Press, 2000.

[17] S. Lengagne, P. Mathieu, A. Kheddar, and E. Yoshida. Generation of dynamic motions under continuous constraints: Efficient computation using b-splines and taylor polynomials. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 698–703, Taipei, Taiwan, 2010.

[18] D. Salomon. *Computer Graphics and Geometric Modeling*. Springer-Verlag, 1999.

[19] M. Vukobratović and B. Borovac. Zero-Moment Point - thirty five years of its life. *International Journal of Humanoid Robotics*, 2004.

[20] S. Nishiwaki, S. Hattori, F. Kanehiro, S. Kajita, and H. Hirukawa. Online mixture and connection of basic motions for humanoid walking control by footprint specification. In *IEEE Int. Conf. on Robotics and Automation*, pages 4110–4115, May 2001.

[21] S. Nakaoka. Choreonoid: Extensible virtual robot environment built on an integrated gui framework. In *IEEE/SICE International Symposium on System Integration*, December 2012.

[22] T. Sugihara and Y. Nakamura. Whole-body cooperative balancing of humanoid robot using cog jacobian. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 2575–2580, 2002.

[23] N. Farkas and R. D. Ramsier. Measurement of coefficient of restitution made easy. *Physics education*, 2006.

[24] AIST Humanoid Research Group. Openhrp3. http://www.openrtp.jp/openhrp3/en/index.html, 2012.

[25] S. Nakaoka, S. Hattori, F. Kanehiro, S. Kajita, and H. Hirokawa. Constraint-based dynamics simulator for humanoid robots with shock absorbing mechanisms. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2007.

[26] R. Cisneros, E. Yoshida, and K. Yokoi. Ball dynamics simulation on openhrp3. In *IEEE Int. Conf. on Robotics and Biomimetics*, pages 871–877, 2012.