2016 IEEE International Conference on Robotics and Automation (ICRA)
Stockholm, Sweden, May 16-21, 2016

Real-Time Planning and Execution of Evasive Motions for a Humanoid Robot

Marco Cognetti, Daniele De Simone, Leonardo Lanari, Giuseppe Oriolo

Abstract— **We present a method for performing evasive motions with a humanoid robot. In the considered scenario, the robot is standing in a workspace, when a moving obstacle (e.g., a human, or another robot) enters its safety area and heads towards it; the humanoid must plan and execute in real-time a maneuver that avoids the collision. The proposed method goes through several conceptual steps. Once the entrance of the moving obstacle in the safety area is detected, its approach direction relative to the robot is determined. On the basis of this information, a suitable evasion maneuver represented by footsteps is generated. From these, an appropriate trajectory is computed for the Center of Mass of the humanoid. Finally, joint motion commands are generated so as to track such trajectory. All computations make use of closed-form expressions and are therefore suitable for real-time implementation. The proposed approach is validated via simulations and experiments on a NAO humanoid. The possibility of adapting the basic method so as to be used in a replanning framework is also investigated.**

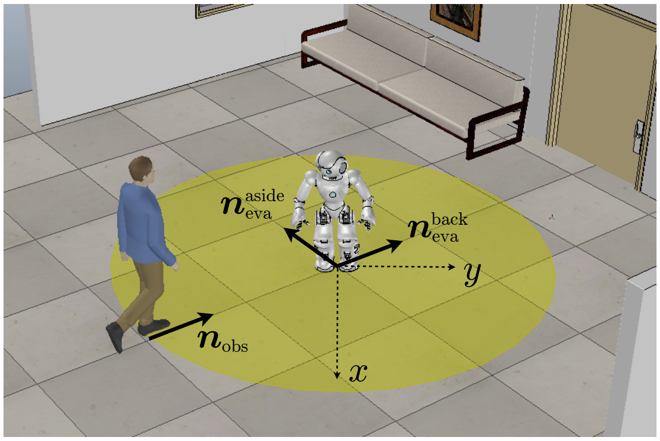
$n_{\text{eva}}^{\text{aside}}$
$n_{\text{eva}}^{\text{back}}$
y
n_{obs}
x

Fig. 1. The situation of interest. A moving obstacle enters the safety area of a humanoid and heads towards it. The humanoid must plan and execute a fast evasive motion. Note the moving frame associated to the humanoid.

I. Introduction

In the last years, robots are making their way out of factories, thanks to the introduction of lightweight, compliant manipulators and the development of new techniques for safe coexistence and interaction with humans. The investigation of these issues has generated many interesting results (e.g., see [1], [2]). Among the available tools, we mention basic layers for detection and avoidance of moving obstacles as well as the definition of safety measures [3].

The same safety problems must be solved for humanoid robots before they are allowed to share an environment with humans. Motivated by a research project (`www.comanoid.eu`) targeting the deployment of humanoids in aeronautic shopfloors, we address here the basic problem of performing real-time evasion maneuvers for a humanoid robot when a moving obstacle enters its vicinity.

The problem of collision avoidance in the presence of moving obstacles has been widely studied in the robotics literature since the 1980's [4]. A number of methods have been proposed (e.g., see [5], [6] and the references therein) for fixed-base or mobile robots, but the case of humanoids has received little attention; existing works include [7], [8], [9]. Indeed, these robots have peculiar features that make the problem more challenging: on the one hand, they are capable of performing evasive motions via steps; on the other, any movement must be planned so as to maintain balance [10]. A closely related problem, which also involves stepping and balancing, is push recovery in humanoid robots. Several authors have investigated this issue; e.g., see [11] or the capture point concept analyzed in [12].

We propose an algorithmic framework which in principle can accommodate (and in which we intend to test) various evasion strategies. The proposed method goes through several conceptual steps. Once the entrance of the moving obstacle in the safety area is detected, its approach direction relative to the robot is determined. From this information, an evasion maneuver represented by footsteps is generated using a controlled unicycle as a reference model. From the footstep sequence, we compute an appropriate trajectory for the Center of Mass of the humanoid, which is finally used to generate joint motion commands that track such trajectory.

In the interest of safety, it is obviously essential that the reaction time (from detection of the moving obstacle to start of the evasive motion) is as small as possible. This is achieved by making use of closed-form expressions throughout the method, and results in an algorithm suitable for real-time implementation. In particular, differently from recently proposed algorithms for on-line generation of humanoid motions [13], [14], we rely on the existence of analytical expressions relating a desired Zero Moment Point trajectory to the associated bounded Center of Mass trajectory, as illustrated in [15].

The paper is organized as follows. In the next section, we formulate the evasion problem and outline our approach, which mainly hinges on two conceptual blocks. The first, discussed in Sect. III, is the generation of an appropriate evasion maneuver in the form of a timed sequence of foot-

The authors are with the Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma, via Ariosto 25, 00185 Roma, Italy. E-mail: {cognetti,desimone,oriolo,lanari}@diag.uniroma1.it.
This work is supported by the EU FP7 project COMANOID.

978-1-4673-8026-3/16/$31.00 ©2016 IEEE

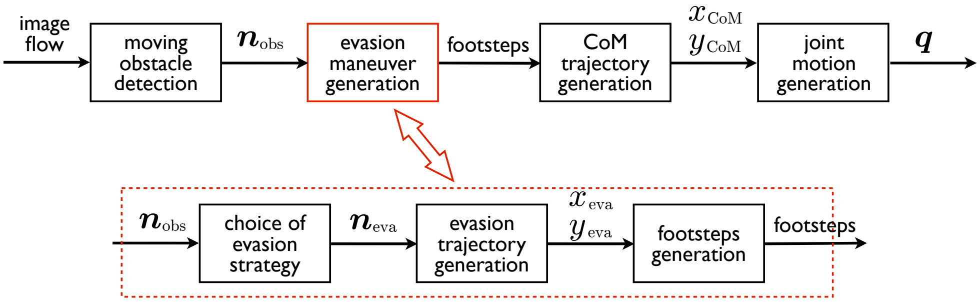

Fig. 2. A block scheme of the proposed approach for planning and executing evasive motions.

steps. The second is the computation of a bounded Center of Mass trajectory associated to these footsteps, as described in Sect. IV. Simulations and experiments on a NAO humanoid are presented in Sect. V. An adaptation of the basic method for use in a replanning framework is discussed in Sect. VI. Future work is mentioned in the concluding section.

II. Formulation and Approach

In this paper, we shall consider the following situation (see Fig. 1). A humanoid robot is standing in a workspace, when a moving obstacle (e.g., a human, or another robot) enters its safety area and heads towards it. If the humanoid does not move, there will be a collision. The humanoid must plan and execute a fast evasive motion to prevent this from happening.

This is a simplified situation under several aspects:

- The humanoid is not executing any particular task; or, if it is, it is ready to abort it and focus on evasion only.
- The obstacle speed is sufficiently low that an evasive motion is possible; moreover, the obstacle does not change its direction. On the other hand, this is the worst possible direction because it aims directly to the humanoid initial position.
- The robot can freely move inside the safety area for performing the evasive motion.

While we intend to remove these assumptions in the future (for example, the hypothesis of constant obstacle direction is relaxed in Sect. VI), the above situation may already be of interest in practical situations.

The proposed approach for planning and executing an evasive motion is summarized in Fig. 2. The first step is to detect the entrance of the moving obstacle in the safety area of the humanoid and to determine its approach direction $\boldsymbol{n}_{\text{obs}}$ relative to the robot (see Fig. 1). The second block reacts to this information by generating a suitable evasion maneuver, represented by footsteps. From these, an appropriate trajectory is computed for the Center of Mass (CoM) of the humanoid. Finally, joint motions are generated so as to track such CoM trajectory, and used to actually move the humanoid, which we assume to be position-controlled.

In this paper, we will not detail the structure of the first block. It is assumed that the robot is equipped with a sensor that provides the necessary information, in particular $\boldsymbol{n}_{\text{obs}}$. In our simulations and experiments, we will use a depth camera on the humanoid head for this purpose. In a similar way, we shall not dwell on the structure of the last block; we assume that joint motion is straightforwardly generated from the planned CoM trajectory by pseudoinverse-based kinematic control (e.g., see [16]). In the following, we shall instead focus on the second and third blocks, i.e., evasion maneuver generation and CoM trajectory generation.

We emphasize the following points.

- All computations should be performed as fast as possible. The reason is obvious: the smaller the reaction time between stimulus (detection of the moving obstacle) and action (beginning of the evasive motion), the more likely is the robot to perform successful collision avoidance. In view of this requirement, we shall look for closed-form solutions and expressions at all stages.
- Every computation in these blocks should be amenable to *replanning* (see Sect. VI).

III. Evasion Maneuver Generation

The Evasion Maneuver Generation block receives as input a unit vector $\boldsymbol{n}_{\text{obs}}$ representing the direction of an incoming obstacle and produces as output a sequence of footsteps that implements an appropriate evasion maneuver. This is achieved by three conceptual steps (Fig. 2): first, an evasion direction $\boldsymbol{n}_{\text{eva}}$ is chosen; then, a planar trajectory that quickly aligns with this direction is generated as a reference for the humanoid; finally, footsteps are appropriately placed around this trajectory. Each step is discussed in detail below.

A. Choice of evasion strategy

Two evasion strategies are considered in this paper:

- *move back:* the humanoid aligns with the direction of the obstacle and moves backwards;
- *move aside:* the humanoid aligns with the direction orthogonal to that of the obstacle and moves backwards.

First, it should be noted that both strategies dictate that the humanoid moves backwards. For the *move back* strategy, this is obvious — moving forward would mean approaching the obstacle. For the *move aside*, this requirement is related to the possibility of keeping the obstacle in view, as moving backwards allows to maintain the obstacle in the half-plane in front of the robot.

The *move back* strategy is obviously aimed at maximizing the distance between the humanoid and the obstacle. In

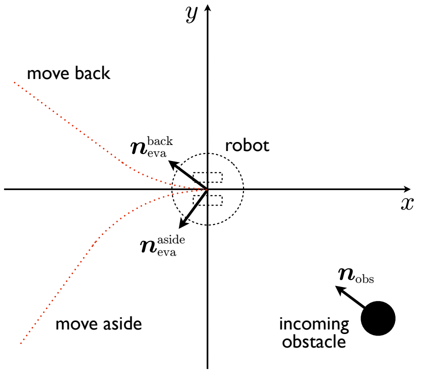
y
move back
$n_{\mathrm{eva}}^{\mathrm{back}}$
robot
x
$n_{\mathrm{eva}}^{\mathrm{aside}}$
n_{obs}
move aside
incoming obstacle

Fig. 3. Generation of the evasion trajectory using a controlled unicycle as reference model.

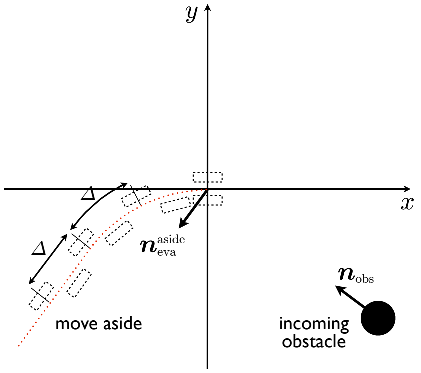

Fig. 4. Footstep placement around the evasion trajectory (a move aside maneuver is considered for illustration).

humans, this is a very instinctive reaction related to an evolutionary rooted tendency called *approach aversion* [17]. However, moving back is not sufficient to avoid collision if the obstacle moves faster than the humanoid.

The *move aside* strategy embodies a different policy, i.e., moving the humanoid as fast as possible away from the course of the obstacle. If executed sufficiently fast, this strategy may allow to avoid obstacles faster than the robot. The downside is that the distance between the humanoid and the obstacle may decrease before increasing again.

In practice, the *move back* strategy is realized by setting $\boldsymbol{n}_{\mathrm{eva}}^{\mathrm{back}} = \boldsymbol{n}_{\mathrm{obs}}$, whereas *move aside* corresponds to $\boldsymbol{n}_{\mathrm{eva}}^{\mathrm{aside}} = \boldsymbol{n}_{\mathrm{obs}}^{\perp}$, where $\boldsymbol{n}_{\mathrm{obs}}^{\perp}$ is the normal unit vector to $\boldsymbol{n}_{\mathrm{obs}}$ in the half-plane behind the robot (see Fig. 1).

B. Evasion trajectory generation

To generate a trajectory that implements the chosen evasion strategy, we use a controlled unicycle as reference model. Studies on human locomotion [18], [19] have identified a gait model in which the orientation of the body is for most of the time tangent to the path. In other words, human trajectories closely resemble those typical of nonholonomic wheeled mobile robots, such as the unicycle. This kind of viewpoint was already effectively assumed in [20].

Refer to Fig. 3 (compare with Fig. 1). The reference unicycle starts at the origin of the humanoid frame, with the same heading, and obeys the well-known model

$$\begin{aligned} \dot{x} &= v \sin\theta && (1)\\ \dot{y} &= v \cos\theta && (2)\\ \dot{\theta} &= \omega, && (3) \end{aligned}$$

where x, y are the unicycle Cartesian coordinates, θ is its orientation w.r.t. the x axis, and v, ω are the driving and steering velocity inputs. If we denote by $\boldsymbol{n}_{\mathrm{eva}}$ the unit vector of the generic evasion direction, and by $\angle \boldsymbol{n}_{\mathrm{eva}} \in [0, 2\pi)$ its phase angle, then a unicycle traveling *backwards* in the direction of $\boldsymbol{n}_{\mathrm{eva}}$ would have orientation $\theta_{\mathrm{eva}} = \angle \boldsymbol{n}_{\mathrm{eva}} - \pi$.

The proposed control law aims at aligning the unicycle with the desired orientation θ_{eva} while traveling at a constant velocity, typically chosen to realize the evasion maneuver as fast as possible. In particular, consider the following feedback control law

$$\begin{aligned} v &= \bar{v} && (4)\\ \omega &= k\ \mathrm{sign}(\theta_{\mathrm{eva}} - \theta), && (5) \end{aligned}$$

where $\bar{v} < 0$ is a constant negative driving velocity and k is a positive constant. The resulting trajectory starts at the origin of the plane and proceeds backwards along an arc of circle of radius $|\bar{v}|/k$, until the desired orientation θ_{eva} is achieved at the finite time instant $t_s = |\theta_{\mathrm{eva}}|/k$. At t_s, the trajectory becomes a line (see Fig. 3). Integration of the model equations (1–3) under the control law (4–5) readily provides a closed form for such trajectory, i.e.,

$$\begin{aligned} x(t) &= \bar{v}\,\frac{\sin kt}{k} && (6)\\ y(t) &= \mathrm{sign}(\theta_{\mathrm{eva}})\,\bar{v}\,\frac{1-\cos kt}{k} && (7)\\ \theta(t) &= \mathrm{sign}(\theta_{\mathrm{eva}})\,kt && (8) \end{aligned}$$

for $t \leq t_s$ and

$$\begin{aligned} x(t) &= x(t_s) + \bar{v}(t - t_s)\cos\theta_{\mathrm{eva}} && (9)\\ y(t) &= y(t_s) + \bar{v}(t - t_s)\sin\theta_{\mathrm{eva}} && (10)\\ \theta(t) &= \theta_{\mathrm{eva}} && (11) \end{aligned}$$

for $t > t_s$.

The availability of a closed form expression for the evasion trajectory is particularly important in view of the adoption of a replanning scheme, because it means that the trajectory can be recomputed in real time if the desired θ_{eva} angle changes.

C. Footstep generation

The final stage of generating an evasion maneuver for the humanoid is the computation of the backward footsteps around the evasion trajectory. The idea is to use a constant stepsize Δ, measured along the trajectory itself (see Fig. 4). This is simply realized by sampling the trajectory expressions (6–8) and (9–11) using a constant time interval $\Delta t = \Delta/|\bar{v}|$, and displacing the x, y, θ samples alternatively to the right and to the left of the trajectory. The first step is an exception, because it is actually a half-step; note also that the inner foot is the first to move.

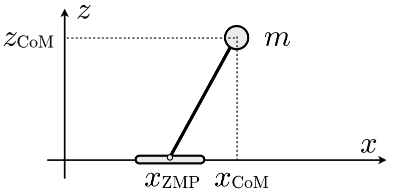

Fig. 5. The Linear Inverted Pendulum (LIP) in the sagittal plane.

IV. CoM Trajectory Generation

We now discuss the CoM Trajectory Generation block of Fig. 2, which receives as input a sequence of timed footsteps and produces as output a CoM trajectory to be fed to the joint motion generation module. In view of our quest for fast computations, using the full humanoid dynamics to this purpose is out of the question. We shall instead resort to the Linear Inverted Pendulum (LIP, see Fig. 5), an approximate model that describes the motion of the humanoid CoM when its height is kept constant and no rotational effects are taken into account. Use of the LIP model for real-time CoM trajectory generation is effective in spite of its simplicity [10]. In particular, we follow the Zero Moment Point (ZMP)-based approach of [15], [21] to achieve closed-form generation of bounded CoM trajectories, a feature which will prove fundamental in the replanning context of Sect. VI.

From the timed footstep sequence, a desired trajectory of the ZMP is derived (e.g., by linear interpolation, as in Fig. 6). The idea is to use the LIP model to identify a bounded CoM trajectory associated to the desired ZMP trajectory.

For the LIP, the lateral (y) and sagittal (x) motions are completely decoupled and governed by identical linear differential equations. In the following, we consider the sagittal motion for illustration; similar formulas for the lateral motion can be immediately derived. Letting $\eta = \sqrt{g/z_{\mathrm{CoM}}}$, where z_{CoM} is the constant height of the CoM, the differential equation[1] describing the sagittal motion is

$$\ddot{x}_{\mathrm{CoM}}(t) - \eta^2(x_{\mathrm{CoM}}(t) - x_{\mathrm{ZMP}}(t)) = 0, \qquad (12)$$

where x_{CoM} and x_{ZMP} are the x-position of the CoM and the ZMP, respectively. This is clearly an unstable dynamics.

Denote by $x^*_{\mathrm{ZMP}}(t)$ the desired trajectory of the ZMP, and let $x_{\mathrm{ZMP}}(t) = x^*_{\mathrm{ZMP}}(t)$ in eq. (12). We are interested in the only bounded solution $x^*_{\mathrm{CoM}}(t)$ of the resulting equation that matches the initial position $x_{\mathrm{CoM}}(0)$. Denoting by

$$x_{\mathrm{u}}(t) = \eta \int_0^{\infty} e^{-\eta\tau} x^*_{\mathrm{ZMP}}(t+\tau) d\tau \qquad (13)$$

$$x_{\mathrm{s}}(t) = \eta \int_0^{t} e^{-\eta(t-\tau)} x^*_{\mathrm{ZMP}}(\tau) d\tau \qquad (14)$$

respectively the unstable and stable contribution of the LIP dynamics, and imposing the *boundedness constraint*

$$x_{\mathrm{u}}(0) = x_{\mathrm{CoM}}(0) + \frac{1}{\eta}\dot{x}_{\mathrm{CoM}}(0), \qquad (15)$$

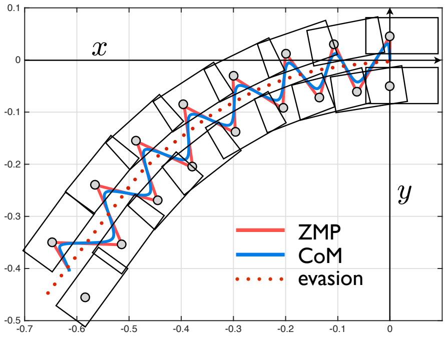

Fig. 6. ZMP-CoM trajectory generation for evasion: geometric path.

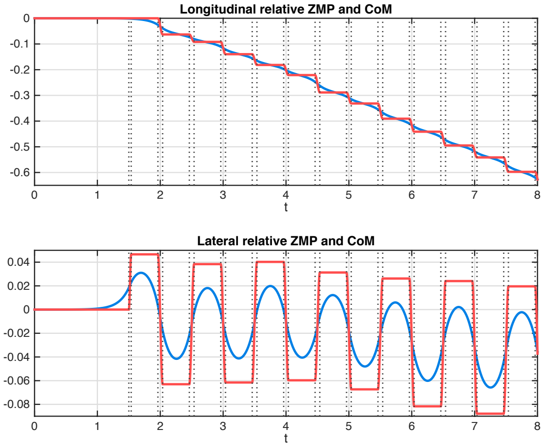

Fig. 7. ZMP-CoM trajectory generation for evasion: time evolution.

the desired bounded solution associated to the reference $x^*_{\mathrm{ZMP}}(t)$ is computed as

$$x^*_{\mathrm{CoM}}(t) = e^{-\eta t} x_{\mathrm{CoM}}(0) + \frac{x_{\mathrm{s}}(t) - e^{-\eta t} x_{\mathrm{u}}(0) + x_{\mathrm{u}}(t)}{2}. \qquad (16)$$

Formulas (13–14) and (16) are valid for any ZMP trajectory. Typical choices of $x^*_{\mathrm{ZMP}}(t)$ result in integrable expressions and, ultimately, in a closed form for $x^*_{\mathrm{CoM}}(t)$.

Note that the unstable contribution $x_{\mathrm{u}}(t)$ is *non-causal* in nature; hence, the computation of $x^*_{\mathrm{CoM}}(t)$ requires knowledge of the future ($> t$) desired evolution of the ZMP.

Once $x^*_{\mathrm{CoM}}(t)$ is computed, it can be used as a reference trajectory for real-time joint motion generation. While $x^*_{\mathrm{CoM}}(0) = x_{\mathrm{CoM}}(0)$ by construction, the initial velocity

$$\dot{x}^*_{\mathrm{CoM}}(0) = \eta\,(x_{\mathrm{u}}(0) - x_{\mathrm{CoM}}(0)) \qquad (17)$$

may not match the true initial condition. This would result in a transient error for the CoM and, thus, for the ZMP. One way to avoid this error (and the associated risk) is to perform an *anticipative* motion aimed at achieving the correct initial velocity before actually starting to track $x^*_{\mathrm{CoM}}(t)$,

We conclude this section by showing in Figs. 6–7 a typical result of the proposed CoM trajectory generation procedure. Here, the reference ZMP trajectory was built by choosing line segments to interpolate the evasion footsteps, then dividing each step into a double and single support

[1] Note that this equation applies to both point feet and finite-sized feet [12].

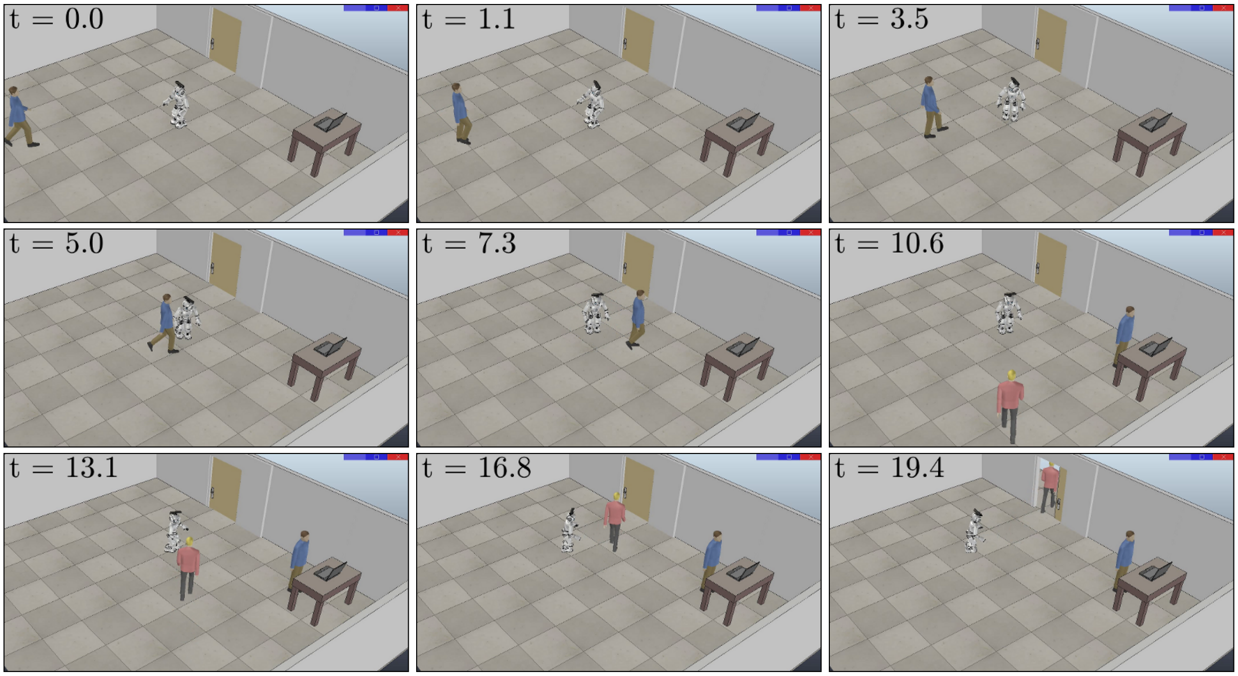

Fig. 8. Evasive motions using the *move aside* strategy: snapshots from a simulation.

phase, and choosing the ZMP to be cubic in time over the first and constant over the second.

V. Simulations and Experiments

The humanoid chosen for implementation was a NAO by Aldebaran Robotics. We have equipped the robot with a depth camera (Asus Xtion PRO Live) mounted on its head. Using this sensor, it can detect obstacles that enter its safety area, defined as a circle of radius 1.5 m, and compute the unit vector $\boldsymbol{n}_{\text{obs}}$ that represents the obstacle approach direction.

The proposed method was first tried in simulation. In particular, we used V-REP for performing full physical simulation (including joint control) of the robot motion: this means that the joint-level commands generated by our method are actually sent to the NAO low-level servos.

The result of a typical simulation is shown in Fig. 8 (see the accompanying video for a clip). The humanoid is standing at the center of a room when a human walks in (1st snapshot), directed towards the desk on the right. As the human enters the safety area of the humanoid (2ns snapshot), the latter detects this fact and generates an evasive motion using the *move aside* strategy. In particular, following the scheme in Fig. 2, an evasion trajectory is first generated using the controlled unicycle model, with $\bar{v} = 0.04$ m/s and $k = 0.2$, and footsteps are placed around this trajectory, using a stepsize $\Delta = 0.08$ m, in the range of NAO capabilities. Then, a ZMP trajectory interpolating the footsteps is computed, with duration of the double and single support respectively at 0.122 s and 0.425 s, and the corresponding bounded CoM trajectory is generated. Finally, joint commands are produced via kinematic control. Overall, the achieved response time (between the detection of the moving obstacle entering the safety area and availability of the first joint command) is around 21 ms, confirming that the use of closed-form expressions in all stages of motion generation makes real-time evasion possible. As a result, the humanoid performs a successful *move aside* evasion (3rd to 5th snapshot). The simulation continues with another human crossing the room (6th snapshot) towards the door, and the robot executing another evasive motion (7th to 9th snapshot).

Experimental validation was performed using two NAOs, one acting as the robot and the other (teleoperated) as the moving human. The result is shown in Fig. 9 (see the accompanying video for a clip). Generation of the evasive motion is performed on-board, using the same parameters of the previous simulation. As expected, a *move aside* evasive motion is successfully executed.

VI. Replanning

Assume now that the moving obstacle is *malicious*, in that it continuously changes its course so as to keep aiming at the humanoid while the latter performs the evasive motion. Depending on the obstacle's own motion model, this may result in different approach trajectories, more or less aggressive. In response to the obstacle behavior, the humanoid should update its evasive motion too. The mechanism for achieving this *replanning* is already embedded in the proposed scheme, and in particular in the unicycle feedback control law (4–5) that generates the evasion trajectory.

A complete study of the evasion trajectories resulting from the interaction of the obstacle approach trajectory with the controlled unicycle model, although interesting, is out of the scope of this paper. In the following, we focus on the algorithmic part of the replanning procedure as a preliminary step in that direction.

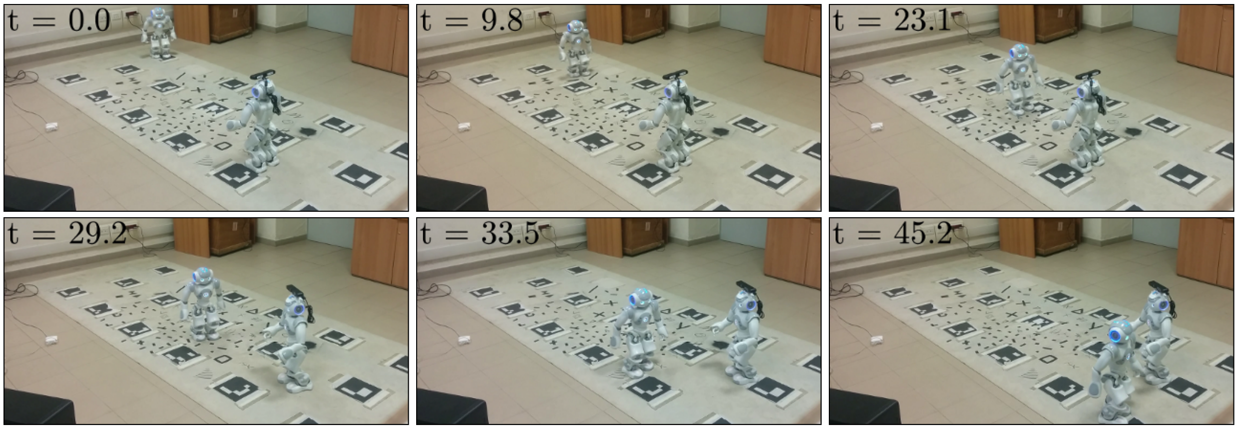

Fig. 9. Evasive motion using the *move aside* strategy: snapshots from an experiment.

A. Replanning the evasion maneuver

A change of direction in the motion of the obstacle can be easily detected by the same sensor that detects its entrance into the humanoid safety area. Once a new value $\boldsymbol{n}'_{\rm obs}$ is available, it is immediately used to compute a new $\boldsymbol{n}'_{\rm eva}$, the associated evasion trajectory given by (6–8) and (9–11) with $\theta'_{\rm eva}$ in place of $\theta_{\rm eva}$, and the corresponding footsteps. These computations are extremely fast because all expressions are available in closed form. Clearly, both the updated trajectory and footsteps start from the current posture of the humanoid.

B. Replanning the CoM trajectory

Once an updated evasion maneuver is available in the form of a new set of footsteps, the corresponding ZMP and CoM trajectories need to be generated as fast as possible. We assume that the humanoid concludes any current double support phase, so that the new sequence of footsteps will be initiated with the following single support phase. The difficulty is that the CoM trajectory associated to the current maneuver will be different from that corresponding to the new maneuver. In the light of Sect. IV, this means that at the switching instant there will be a mismatch in the CoM initial velocity, and therefore a transient error will be present both in the CoM and in the resulting ZMP. Clearly, as the humanoid is already moving, performing an anticipative motion to match the initial CoM velocity is not an option.

A way to solve this issue consists in introducing some free parameters in the desired ZMP trajectory $x^*_{\rm ZMP}(t)$. These can be used to satisfy the boundedness constraint (15) through $x_{\rm u}(0)$, which depends upon $x^*_{\rm ZMP}(t)$, leaving $x_{\rm CoM}(0)$ and $\dot{x}_{\rm CoM}(0)$ free. This approach can be seen as a simultaneous ZMP-CoM generation along the same lines of [11], [22].

In the following, we illustrate this concept by assuming, for simplicity, a point foot situation (hence no double support). The ZMP is then a sequence of Heaviside functions $u_{\rm step}(t)$ with amplitudes α_i equal to the step lengths; for n steps the reference ZMP becomes

$$x^*_{\rm ZMP}(t) = \sum_{i=1}^{n} \alpha_i u_{\rm step}(t - t_i). \tag{18}$$

With this simple choice we have

$$x_{\rm u}(0) = \sum_{i=1}^{n} \alpha_i e^{-\eta t_i} \tag{19}$$

Considering the presence of free parameters in the first step only, the boundedness constraint can be written as

$$\alpha_1 e^{-\eta t_1} = -\sum_{i=2}^{n} \alpha_i e^{-\eta t_i} + x_{\rm CoM}(0) + \frac{1}{\eta}\dot{x}_{\rm CoM}(0)$$

which shows that we can choose as design parameter either the step length α_1 or its duration t_1. The corresponding solutions are in the first case

$$\alpha_1 = e^{\eta t_1}\left(x_{\rm CoM}(0) + \frac{\dot{x}_{\rm CoM}(0)}{\eta} - \sum_{i=2}^{n} \alpha_i e^{-\eta t_i}\right)$$

and in the second

$$t_1 = -\frac{1}{\eta}\log\frac{1}{\alpha_1}\left(x_{\rm CoM}(0) + \frac{\dot{x}_{\rm CoM}(0)}{\eta} - \sum_{i=2}^{n} \alpha_i e^{-\eta t_i}\right)$$

For illustration, consider the replanning example shown in Fig. 10. Here, the top plot shows a bounded CoM trajectory computed as in Sect. IV from a ZMP trajectory that interpolates an updated sequence of footsteps with fixed timing and lengths. At the switching instant $t = 0$, the nominal CoM velocity (derived from the boundedness constraint) is $\dot{x}^*_{\rm CoM}(0) = -0.108$ m/s. Now assume that the actual CoM velocity, resulting from the previous plan, is -0.08 m/s. Tracking the above nominal CoM trajectory with mismatched initial conditions would introduce a transient error. As an alternative, we can introduce a free parameter in the ZMP trajectory, namely in the first step, and compute a matched CoM trajectory as shown above. If the duration of the first step is maintained as in the nominal plan, then a longer step length is obtained (center plot), whereas if the length of the first step is maintained, a shorter step is obtained (bottom plot); these outcomes are consistent with the observation that the actual CoM velocity is larger than needed.

This example shows that choosing as free parameter the first step length actually leads to modifying the footstep

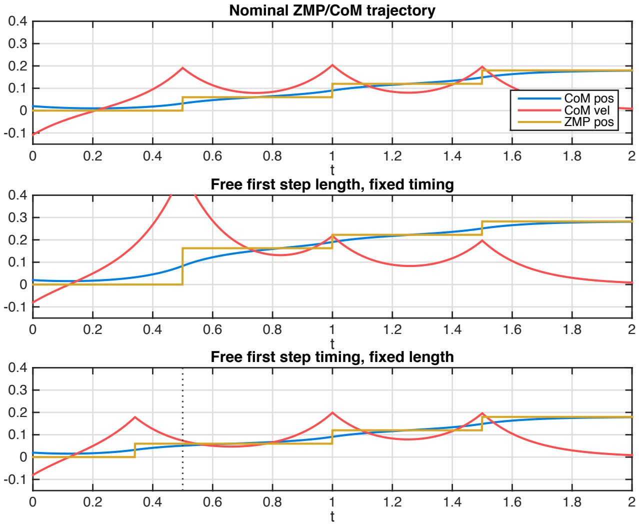

Fig. 10. A replanning example. Top: the nominal CoM trajectory corresponding to an updated evasion maneuver is mismatched w.r.t. the CoM initial velocity. Center: a matched CoM trajectory computed by freeing the first step length. Bottom: a matched CoM trajectory computed by freeing the first step duration.

sequence (at least the first step) with respect to the updated sequence coming from the Evasion Maneuver Generation block. For this reason, and also in view of the fact that the variable length approach may lead to high CoM velocities (center plot in Fig. 10), the variable duration approach appears to the most convenient for replanning the CoM trajectory without modifying the updated evasion maneuver.

VII. Conclusions

We have described an approach for real-time planning and execution of evasive motions of a humanoid robot. The proposed method goes through a sequence of conceptual steps. Once the entrance of a moving obstacle in the safety area is detected, its approach direction relative to the robot is determined. On the basis of this information, a suitable evasion maneuver represented by footsteps is generated. From these, an appropriate trajectory is computed for the Center of Mass of the humanoid. Finally, joint motion commands are generated so as to track such trajectory. All these computations make use of closed-form expressions and are therefore suitable for real-time implementation. The proposed approach has been successfully validated via simulations and experiments on a NAO humanoid. The possibility of adapting the basic method so as to be used in a replanning framework has also been discussed.

Future work will address several points, such as:

- development of a *gazing* strategy for faster detection of obstacles entering the robot safety area;
- design and test of additional evasion strategies in addition to the basic ones considered here;
- refinement of the replanning scheme for the case of finite-sized feet and its experimental validation.

References

[1] A. De Luca and F. Flacco, "Integrated control for phri: Collision avoidance, detection, reaction and collaboration," in *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS EMBS International Conference on*, 2012, pp. 288–295.

[2] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1726 – 1743, 2013.

[3] B. Lacevic, P. Rocco, and A. Zanchettin, "Safety assessment and control of robotic manipulators using danger field," *Robotics, IEEE Transactions on*, vol. 29, no. 5, pp. 1257–1270, 2013.

[4] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 500–505.

[5] E. Yoshida and F. Kanehiro, "Reactive robot motion using path replanning and deformation," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 5456–5462.

[6] S. M. Khansari-Zadeh and A. Billard, "A dynamical system approach to realtime obstacle avoidance," *Autonomous Robots*, vol. 32, no. 4, pp. 433–454, 2012.

[7] P. Michel, J. Chestnutt, J. Kuffner, and T. Kanade, "Vision-guided humanoid footstep planning for dynamic environments," in *2005 IEEE-RAS Int. Conf. on Humanoid Robots*, 2005, pp. 13–18.

[8] O. Khatib, L. Sentis, and J.-H. Park, *European Robotics Symposium 2008*. Springer Berlin Heidelberg, 2008, ch. A Unified Framework for Whole-Body Humanoid Robot Control with Multiple Constraints and Contacts, pp. 303–312.

[9] E. Spyrakos-Papastavridis, D. Kanoulas, N. G. Tsagarakis, and D. G. Caldwell, "Optically-regulated impedance-based balancing for humanoid robots," in *2015 IEEE-RAS Int. Conf. on Humanoid Robots*, 2015, pp. 41–45.

[10] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, *Introduction to Humanoid Robotics*. Springer Publishing Company Inc., 2014.

[11] M. Morisawa, K. Harada, S. Kajita, K. Kaneko, J. Sola, E. Yoshida, N. Mansard, K. Yokoi, and J.-P. Laumond, "Reactive stepping to prevent falling for humanoids," in *2009 IEEE International Conference on Humanoid Robots*, 2009, pp. 528–534.

[12] T. Koolen, T. de Boer, J. Rebula, A. Goswami, and J. Pratt, "Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models," *Int. J. of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, 2012.

[13] K. Nishiwaki and S. Kagami, "Online walking control system for humanoids with short cycle pattern generation," *Int. J. of Robotics Research*, vol. 28, no. 6, pp. 729–742, 2009.

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

[15] L. Lanari, S. Hutchinson, and L. Marchionni, "Boundedness issues in planning of locomotion trajectories for biped robots," in *2014 IEEE-RAS Int. Conf. on Humanoid Robots*, 2014, pp. 951–958.

[16] M. Cognetti, P. Mohammadi, G. Oriolo, and M. Vendittelli, "Task-oriented whole-body planning for humanoids based on hybrid motion generation," in *2014 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2014, pp. 4071–4076.

[17] C. K. Hsee, Y. Tu, Z. Y. Lu, and B. Ruan, "Approach aversion: Negative hedonic reactions toward approaching stimuli," *Journal of Personality and Social Psychology*, vol. 106, no. 5, pp. 699–712, 2014.

[18] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion – an inverse optimal control approach," *Autonomous Robots*, vol. 28, pp. 369–383, 2010.

[19] T.-V.-A. Truong, D. Flavigne, J. Pettre, K. Mombaur, and J.-P. Laumond, "Reactive synthesizing of human locomotion combining nonholonomic and holonomic behaviors," in *3rd IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2010, pp. 632–637.

[20] A. Faragasso, G. Oriolo, A. Paolillo, and M. Vendittelli, "Vision-based corridor navigation for humanoid robots," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 3190–3195.

[21] L. Lanari and S. Hutchinson, "Inversion-based gait generation for humanoid robots," in *2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2015.

[22] K. Nishiwaki and S. Kagami, "Simultaneous planning of com and zmp based on the preview control method for online walking control," in *2011 IEEE International Conference on Humanoid Robots*, 2011, pp. 745–751.