



HAL
open science

Model predictive control for tilt recovery of an omnidirectional wheeled humanoid robot

Jory Lafaye, Cyrille Collette, Pierre-Brice Wieber

► **To cite this version:**

Jory Lafaye, Cyrille Collette, Pierre-Brice Wieber. Model predictive control for tilt recovery of an omnidirectional wheeled humanoid robot. IEEE International Conference on Robotics and Automation (ICRA), May 2015, Seattle, United States. pp.5134-5139, 10.1109/ICRA.2015.7139914. hal-02487657

HAL Id: hal-02487657

<https://inria.hal.science/hal-02487657>

Submitted on 21 Feb 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Model Predictive Control for Tilt Recovery of an Omnidirectional Wheeled Humanoid Robot

Jory Lafaye, *Aldebaran Robotics, jlafaye@aldebaran-robotics.com*
Cyrille Collette, *Aldebaran Robotics, ccollette@aldebaran-robotics.com*
Pierre-Brice Wieber, *Inria Grenoble Rhône-Alpes, France*

Abstract—The goal of this paper is to present a real-time controller for an omnidirectional wheeled humanoid robot which can be strongly disturbed and tilt around its wheels. It is based on two linear model predictive controllers, managed by a tilt supervisor, which detects changes of the dynamic model caused by the tilt of the robot. Experimental results are proposed on the robot Pepper made by Aldebaran, showing good performance in term of stability and robustness.

Index Terms—Humanoid Robotics, Mobile Robotics, Omnidirectional Wheeled Robot, Tilt Recovery, Push Recovery, Linear Model Predictive Control, Quadratic Programming, Aldebaran Pepper Robot

I. INTRODUCTION

Pepper is a humanoid robot standing on a holonomic mobile base, equipped with three omnidirectional wheels. It is developed by Aldebaran Robotics for the Japanese Telecommunication Company Soft Bank, to welcome customers in their shops. Its standard posture is to have all three wheels firmly in contact with the ground, as shown in Fig. 1. In this case, dynamic balance is ensured if the Center of Pressure (CoP) lies inside the support triangle outlined by the three wheels. A Model Predictive Control (MPC) method has been proposed in [1] to control the motion of Pepper in such situations. In the absence of disturbance, this control law manages to continuously maintain the balance of the robot and keep all three wheels firmly in contact with the ground, even when realizing sharp turns and other high speed motions.

Strong disturbances may upset this situation, however, and the robot can begin to tilt, and fall if appropriate actions are not undertaken. The problem, when tilting occurs, is that the three wheels of the robot are not all in contact with the ground anymore: as a result, the dynamics of the robot changes significantly and the control law proposed in [1] is not appropriate anymore. We propose in this paper to tackle this problem of controlling the humanoid robot Pepper over these different phases, when the robot is standing firmly on its three wheels, and when it is tilting with only two wheels on the ground, with significantly different dynamical behaviors.

The problem of balancing robots on two wheels, similarly to inverted pendulums, has already been studied extensively [2], [3], [4], [5], [6]. But in all those cases, the objective is to maintain long term balance on two wheels, keeping the Center of Mass (CoM) of the robot in equilibrium above



Fig. 1. Pepper is a humanoid robot standing on a holonomic mobile base, equipped with three omnidirectional wheels. It is developed by Aldebaran Robotics for the Japanese Telecommunication Company Soft Bank, to welcome customers in their shops.

the two wheels, whereas in our case, the objective is to recover a firm stance on three wheels as quickly as possible. This is a very different control objective.

Controlling the tilt of vehicles equipped with three wheels has already been discussed in [7], [8], [9], [10]. But in these cases, all three wheels are supposed to always stay firmly in contact with the ground, and tilting refers in fact to simply controlling the motion of the CoM above the wheels, in a way very similar to what was done in [1] for Pepper when it is not tilting.

In the case of biped humanoid robots, strong disturbances can be absorbed by adapting the motion of the CoM, or by realizing adequate steps [11], [12], [13], [14], [15], [16], [17]. But in both cases, the dynamics of the biped robots is not particularly changing. Even though the global behavior of Pepper looks similar, adapting the motion of its CoM or moving its “foot” (the mobile base) on the ground, the underlying change of dynamical behavior makes it a significantly different problem.

In the end, it appears that the problem of tilt recovery for an omnidirectional wheeled robot such as Pepper has never really been approached in the robotics literature. Having to deal with different phases of motion, with significantly different dynamical behaviors, we propose to develop sep-

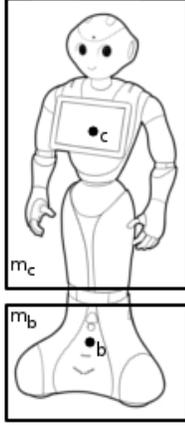


Fig. 2. When the robot is standing firmly on its three wheels and not tilting, the model proposed in [1] considers separately the motion of the mobile base (with CoM b) and the motion of the upper body (with CoM c)

arate control laws for each phases, with a supervisor to decide which one should be used, depending on the situation. One difficulty is that the robot Pepper is not equipped with force sensors. As a result, decisions must be based only on wheel velocity, joint motion and informations obtained from the Inertial Measurement Unit (IMU) composed of three accelerometers and three gyrometers. A phase estimator is required therefore to make appropriate decisions.

No assumptions are made on the disturbances, their amplitude, duration or bandwidth, and our solution has been observed to be quite robust to various forms of disturbances. In case the disturbances are too strong and balance cannot be recovered, a fall manager strategy has been implemented in order to protect the robot, by minimizing the impact of the body with the ground, but this is outside of the scope of this paper.

II. DYNAMICAL MODEL WITH THREE WHEELS ON THE GROUND

When the robot is standing firmly on its three wheels and not tilting, the model proposed in [1] considers separately the motion of the mobile base (with CoM b) and the motion of the upper body (with CoM c), as shown in Fig. II. With the help of Newton and Euler equations (more details can be found in [1]), the dynamics of this system can take the following form:

$$\begin{aligned} \dot{L}_b + \dot{L}_c = (p - b) \times m_b(\ddot{b} + g) \\ + (p - c) \times m_c(\ddot{c} + g). \end{aligned} \quad (1)$$

with g the gravity acceleration, m_b and m_c the total mass of the mobile base and upper body, \dot{L}_b and \dot{L}_c their angular momenta, and p the CoP of the contact forces on the ground.

The CoM of the mobile base normally has a constant height $b^z = l$ above the ground. Vertical motion of the CoM of the upper body can usually be neglected, so it is usually considered to have a constant height as well: $c^z = h$. Finally, the angular momenta can also be generally neglected: $\dot{L}_b + \dot{L}_c = 0$. With these assumptions, the dynamics (1) of

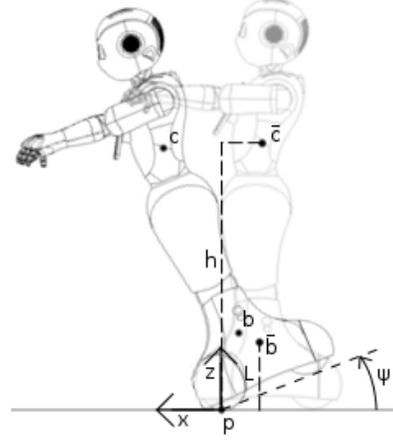


Fig. 3. When the robot tilts, the CoP p lies on the tilting axis. In order to simplify notations, we consider that tilting occurs around two wheels in the direction of the x axis, with an angle $\psi \geq 0$.

the robot can be expressed as a linear relationship between the position of the CoP p on the ground and the horizontal motion of the two CoM b and c :

$$p^{xy} = \frac{m_b b^{xy} + m_c c^{xy}}{m_b + m_c} - \frac{m_b l \ddot{b}^{xy} + m_c h \ddot{c}^{xy}}{(m_b + m_c)g}. \quad (2)$$

Further details and discussion of this model can be found in [1].

III. DYNAMICAL MODEL WITH TWO WHEELS ON THE GROUND

When the robot tilts, the CoP lies on the tilting axis. In order to simplify the following notations, let us consider that tilting occurs around two wheels in the direction of the x axis. Due to the mechanical structure of the robot, we do not consider the possibility that it tilts around a single wheel. Let $\psi \geq 0$ be the tilt angle. Introducing \bar{b} and \bar{c} , the positions of the two CoM if tilting didn't occur, a simple rotation of angle ψ around the tilting axis (where the CoP p lies) gives:

$$\begin{cases} b^x = p^x + (\bar{b}^x - p^x) \cos(\psi) + l \sin(\psi), \\ b^z = l \cos(\psi) - (\bar{b}^x - p^x) \sin(\psi). \end{cases} \quad (3)$$

Linearizing around $\psi = 0$, and neglecting the change of height $(\bar{b}^x - p^x)\psi$ with respect to l , we obtain that

$$\begin{cases} b^x = \bar{b}^x + l\psi, \\ b^z = l. \end{cases} \quad (4)$$

The same can be done for the CoM of the upper body c .

In this case, the linear relationship (2) can be rewritten along the x axis as follows:

$$\begin{aligned} p^x = \frac{m_b(\bar{b}^x + l\psi) + m_c(\bar{c}^x + h\psi)}{m_b + m_c} \\ - \frac{m_b l(\ddot{\bar{b}}^x + l\ddot{\psi}) + m_c h(\ddot{\bar{c}}^x + h\ddot{\psi})}{(m_b + m_c)g}. \end{aligned} \quad (5)$$

Since the CoP lies on the tilting axis, which is at a constant distance d to the center of the mobile base, we have

$$p^x = \bar{b}^x + d. \quad (6)$$

We can reorganize then the previous equation to obtain a linear relationship between variations of the tilt angle and motion of the two CoM:

$$(m_b l + m_c h) \psi - \left(m_b \frac{l^2}{g} + m_c \frac{h^2}{g} \right) \ddot{\psi} = (m_b + m_c) d - m_c (\bar{c}^x - \bar{b}^x) + m_b \frac{l}{g} \ddot{\bar{b}}^x + m_c \frac{h}{g} \ddot{\bar{c}}^x. \quad (7)$$

IV. MODEL PREDICTIVE CONTROL OF THE TILTING MOTION

A. Sampled dynamics and prediction

In order to design the controller of the robot when it is tilting on two wheels, we propose to follow the same design principles as previously used in [1] when the robot is standing firmly on its three wheels, and not tilting. In order to generate smooth motions of the robot, we consider that the CoM of the mobile base and the CoM of the upper body follow a third order dynamics, with a sampling period T :

$$\hat{c}_k^x = \begin{bmatrix} \hat{c}_k^x \\ \dot{\hat{c}}_k^x \\ \ddot{\hat{c}}_k^x \end{bmatrix} = \begin{bmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \hat{c}_{k-1}^x + \begin{bmatrix} \frac{T^3}{6} \\ \frac{T^2}{2} \\ T \end{bmatrix} \ddot{c}_{k-1}^x, \quad (8)$$

the same for \hat{b}_k^x . We will consider the motion of these two CoM over a horizon of N future samples. Introducing $C^x = [\hat{c}_1^x \dots \hat{c}_N^x]^t$, the same for $\dot{C}^x, \ddot{C}^x, B^x, \dot{B}^x, \ddot{B}^x$, we can iterate the third order dynamics (8) to obtain

$$C^x = U_c \ddot{C}^x + S_c \hat{c}_0^x, \quad (9)$$

$$\dot{C}^x = U_{\dot{c}} \ddot{C}^x + S_{\dot{c}} \hat{c}_0^x, \quad (10)$$

$$\ddot{C}^x = U_{\ddot{c}} \ddot{C}^x + S_{\ddot{c}} \hat{c}_0^x. \quad (11)$$

More details can be found in [1].

We propose to do the same with the tilt angle ψ and consider N future samples in $\Psi, \dot{\Psi}, \ddot{\Psi}, \ddot{\Psi}$, with

$$\Psi = U_{\psi} \ddot{\Psi} + S_{\psi} \hat{\psi}_0, \quad (12)$$

$$\dot{\Psi} = U_{\dot{\psi}} \ddot{\Psi} + S_{\dot{\psi}} \hat{\psi}_0, \quad (13)$$

$$\ddot{\Psi} = U_{\ddot{\psi}} \ddot{\Psi} + S_{\ddot{\psi}} \hat{\psi}_0. \quad (14)$$

The linear relationship (7) can then be considered for all N samples at once:

$$(m_b l + m_c h) \Psi - \left(m_b \frac{l^2}{g} + m_c \frac{h^2}{g} \right) \ddot{\Psi} = (m_b + m_c) d - m_c (C^x - B^x) + m_b \frac{l}{g} \ddot{B}^x + m_c \frac{h}{g} \ddot{C}^x \quad (15)$$

and expressed as a function of $\ddot{\Psi}$ with the help of (12)-(14):

$$\begin{aligned} & \left((m_b l + m_c h) U_{\psi} - \left(m_b \frac{l^2}{g} + m_c \frac{h^2}{g} \right) U_{\ddot{\psi}} \right) \ddot{\Psi} = \\ & (m_b + m_c) d - m_c (C^x - B^x) + m_b \frac{l}{g} \ddot{B}^x + m_c \frac{h}{g} \ddot{C}^x \\ & - \left((m_b l + m_c h) S_{\psi} - \left(m_b \frac{l^2}{g} + m_c \frac{h^2}{g} \right) S_{\ddot{\psi}} \right) \hat{\psi}_0. \quad (16) \end{aligned}$$

The matrix on the left-hand side is lower triangular, with a constant

$$(m_b l + m_c h) \frac{T^3}{6} - \left(m_b \frac{l^2}{g} + m_c \frac{h^2}{g} \right) T \quad (17)$$

on the diagonal. It is invertible if this constant is different from zero, that is, for $T > 0$, if

$$T \neq \sqrt{6 \frac{m_b \frac{l^2}{g} + m_c \frac{h^2}{g}}{m_b l + m_c h}}. \quad (18)$$

In the case of Pepper, this corresponds to $T \neq 542$ ms.

In this case, equation (16) can be solved, to give the third derivative of the tilting angle $\ddot{\Psi}$ as a function of the horizontal position and acceleration of the two CoM, $B^x, \dot{B}^x, C^x, \dot{C}^x$. Replacing $\ddot{\Psi}$ in (12)-(14) and using (9)-(11), we can finally obtain the tilting angle, speed and acceleration as a function of the third-derivative of the motion of the two CoM, \ddot{B}^x and \ddot{C}^x . In synthetic form,

$$\Psi = U_{\psi b} \ddot{B}^x + U_{\psi c} \ddot{C}^x + S_{\psi b} \hat{b}_0^x + S_{\psi c} \hat{c}_0^x + S_{\psi} \hat{\psi}_0, \quad (19)$$

$$\dot{\Psi} = U_{\dot{\psi} b} \ddot{B}^x + U_{\dot{\psi} c} \ddot{C}^x + S_{\dot{\psi} b} \hat{b}_0^x + S_{\dot{\psi} c} \hat{c}_0^x + S_{\dot{\psi}} \hat{\psi}_0, \quad (20)$$

$$\ddot{\Psi} = U_{\ddot{\psi} b} \ddot{B}^x + U_{\ddot{\psi} c} \ddot{C}^x + S_{\ddot{\psi} b} \hat{b}_0^x + S_{\ddot{\psi} c} \hat{c}_0^x + S_{\ddot{\psi}} \hat{\psi}_0. \quad (21)$$

B. Control objectives and constraints

Now, our control problem is to find which motion of the mobile base and upper-body CoM b and c will help reducing the tilting angle ψ to zero. In order to reduce impacts on the ground and the risk to bounce back, it is important to reduce also the tilting velocity $\dot{\psi}$ to zero. Our control objective therefore is to minimize the norm of the tilting angle

$$O_1 = \frac{1}{2} \|\Psi\|^2, \quad (22)$$

and the norm of its velocity

$$O_2 = \frac{1}{2} \|\dot{\Psi}\|^2. \quad (23)$$

It is interesting to introduce also a small regularization term, minimizing

$$O_3 = \frac{1}{2} \|\ddot{B}^x\|^2 + \frac{1}{2} \|\ddot{C}^x\|^2. \quad (24)$$

for numerical reasons, but also to ensure the stability of the control law in all situations.

Constraints on the motion of the robot are the following. First of all, there is a maximum speed and acceleration of the mobile base:

$$\begin{cases} -\dot{b}_{\max}^x \leq \dot{B}^x \leq \dot{b}_{\max}^x, \\ -\ddot{b}_{\max}^x \leq \ddot{B}^x \leq \ddot{b}_{\max}^x. \end{cases} \quad (25)$$

The relative motion between the mobile base and the upper body CoM is constrained by the joint limits in the upper body:

$$-k_{\max}^x \leq C^x - B^x \leq k_{\max}^x, \quad (26)$$

considering a maximal distance k_{\max}^x . Finally, the tilting angle is (naturally) always positive:

$$\Psi \geq 0. \quad (27)$$

With the help of equations (9)-(11) and (19)-(21), all these control objectives and constraints can be expressed as functions of the third-derivative of the motion of the two CoM, $[\ddot{B}^x \ \ddot{C}^x]^t$, and aggregated in a simple Quadratic Program (QP) with linear constraints, which can be solved efficiently.

V. SUPERVISING THE DIFFERENT PHASES OF THE MOTION

A. Different phases and their controllers

We have seen that the motion of the robot can go through different phases, depending on whether it is firmly standing on its three wheels, or tilting around two of its wheels. When standing on three wheels (the “no tilt” state in Fig. V), the MPC scheme proposed in [1] is used, and when tilting around two wheels (the “tilt” state in Fig. V), the MPC scheme described in the previous Section is used. A problem is, the transition from the tilting phase, back to a non-tilting phase, can be easily disturbed because of impacts and imprecise detection of the contact state, since the robot is not equipped with force sensors.

As a result, it is important to introduce a third, transition phase (the “landing” state in Fig. V), when the wheel that was in the air during the tilting phase, is approaching the ground. In this case, the same MPC scheme as when standing on three wheels is used, simply waiting for gravity to finish pulling the wheel in the air back to the ground. But with only two wheels on the ground, the mobile base can not be completely controlled. As a result, the trajectory tracking is temporarily disengaged, and the motion of the mobile base on the ground is constrained instead to be parallel to the tilting direction.

B. An impact estimator

In order to decide when and how to transition between these different phases, we need to be able to estimate whether the active tilt recovery controller of the previous Section is required, or whether the current tilt angle and velocity will naturally transition back to the robot having three wheels on the ground, so that the smoother “landing” controller has to be preferred.

If the tilt angle of the mobile base is only affected by gravity, it would have an acceleration

$$\ddot{\psi} = -\frac{g}{l} \cos\left(\psi + \tan^{-1}\left(\frac{l}{d}\right)\right). \quad (28)$$

Considering that this acceleration is approximately constant, the evolution of the tilt angle from an initial angle ψ_0 and velocity $\dot{\psi}_0$ would be

$$\psi(t) = \psi_0 + \dot{\psi}_0 t - \frac{g}{2l} \cos\left(\psi_0 + \tan^{-1}\left(\frac{l}{d}\right)\right) t^2. \quad (29)$$

Now, if

$$\Delta = \dot{\psi}_0^2 + \frac{2g}{l} \cos\left(\psi_0 + \tan^{-1}\left(\frac{l}{d}\right)\right) \psi_0 < 0, \quad (30)$$

or if $\psi_0 \geq \frac{\pi}{2} - \tan^{-1}\left(\frac{l}{d}\right)$ and $\dot{\psi}_0 > 0$, then the tilt angle will not go back to zero naturally, and the controller presented

in the previous Section must be activated to recover balance. If the tilt angle naturally goes back to zero, it will with a velocity at touchdown

$$\dot{\psi}_{td} = -\sqrt{\Delta}. \quad (31)$$

If this velocity is beyond a safety limit, the controller presented in the previous Section must once again be activated, in order to reduce the impact velocity.

C. Transitions

The standard posture for the robot is to stand firmly on its three wheels. When a tilt angle and velocity is detected by the IMU, two different transitions can happen. Either the impact estimator considers that the robot should simply switch to the “landing” controller, and the tilt angle will naturally vanish with limited impact, or the robot must switch to the active tilt recovery controller.

When tilting around two wheels, the active tilt recovery controller is activated, until the impact estimator considers that the robot should safely switch to the “landing” controller, and the tilt angle will naturally vanish, with limited impact.

When the “landing” controller is active, the robot simply waits to stabilize with a firm stance on all three wheels before switching back to the standard controller presented in [1].

VI. IMPLEMENTATION

The controller and the supervisor presented in the previous sections have been implemented on Pepper and validated experimentally in various situations. In this robot, the wheels of the mobile base are controlled in velocity, while the joints of the upper body are controlled in position.

But the controller is designed in the previous section around a third order dynamics, in order to generate smooth motion. The speed of the mobile base is therefore extracted and directly used as a control input for the wheels, while the speed of the CoM of the upper body is transmitted to a standard Inverse Kinematics algorithm and transformed into reference joint positions.

The current state of the robot (\hat{c}_0 , \hat{b}_0 , $\hat{\psi}_0$) is observed in the following way. The angular velocity $\dot{\psi}_0$ of the base is directly measured with gyrometers, and differentiated numerically to obtain acceleration. The tilt angle is obtained with the observer described in [18]. A standard Direct Kinematics algorithm is used then to obtain the whole state of the mobile base and upper body, with the help of joint position sensors and wheel velocity sensors.

VII. EXPERIMENTS

In this section, we are going to present the most simple situation, when the robot is standing still and suddenly pushed from behind with varying strength.

The recovery behavior from a small push can be seen in Fig. VI. In this case, the supervisor does not decide to activate the controller presented in this paper, and prefers instead to recover passively from the disturbance, with the help of gravity. We can observe here how the estimated

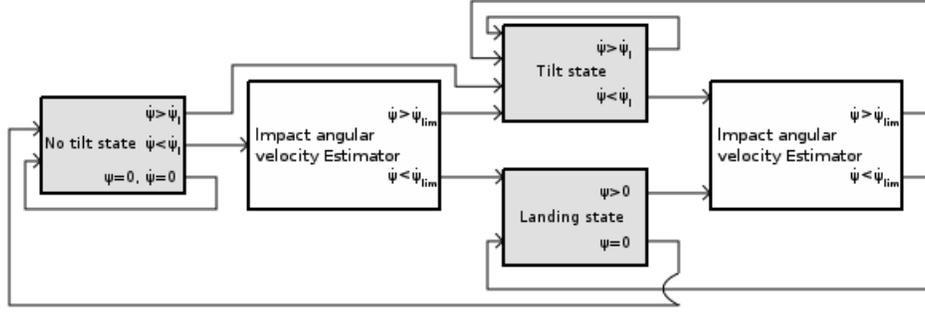


Fig. 4. Supervision of the different phases of the motion and their transitions. If the tilt velocity $\dot{\psi}$ is beyond a limit $\dot{\psi}_l$, the controller transitions directly to the “tilt” state. Below this limit, the choice between the “tilt” state and the “landing” state depends on the impact velocity limit $\dot{\psi}_{lim}$.

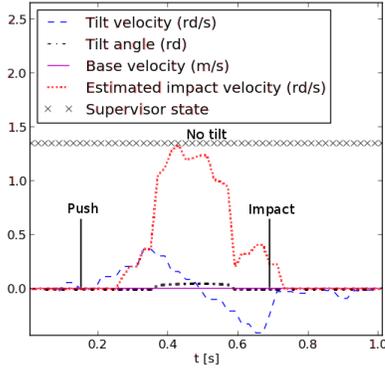


Fig. 5. Experiment 1: Small push recovery experiment. The robot stay in non tilt state

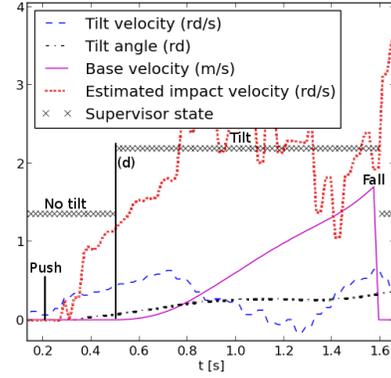


Fig. 7. Strong push recovery experiment. (d) The robot enter in tilt state

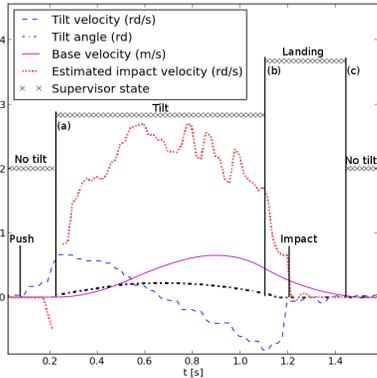


Fig. 6. Medium push recovery experiment. (a) The robot enter in tilt state. (b) The robot enter in landing state. (c) The robot returns in non tilt state

impact velocity is varying, depending on how the disturbance is evolving.

The recovery behavior from a medium push can be seen in Fig. VI. At time $t = 0.21$ s, the impact velocity estimator triggers a transition to the controller presented in this paper, and the velocity of the mobile base starts increasing to compensate for the disturbance. This controller is deactivated at time $t = 1.1$ s, when the estimated impact velocity goes below a threshold at $1.8 \text{ rad}\cdot\text{s}^{-1}$, and the supervisor enters the “landing” state, using only gravity to safely fall back on all three wheels. The “landing” state is exited at time

$\dot{\psi}_d$ (rd/s)	t_d (s)	l_r (m)	\dot{b}_m (m/s)	ψ_m (rd)	$\dot{\psi}_i$ (rd/s)
Weak pushes					
0.30	0.2	0	0	0.06	0.38
0.44	0.4	0.06	0.20	0.16	0.25
0.45	0.2	0	0	0.12	0.42
Medium pushes					
0.59	0.2	0.27	0.51	0.20	0.48
0.60	0.4	0.45	0.56	0.24	0.55
0.80	0.2	0.32	0.77	0.21	0.80
Strong pushes					
0.58	0.5	0.85	1.4	0.28	0.81
0.61	0.7	1.2	1.5	0.32	0.83
0.78	0.5	1.0	1.8	0.29	0.80

Fig. 8. Table quantifying the push recovery performance for different disturbances. $\dot{\psi}_d$ is the angular velocity at the end of the disturbance, t_d is the disturbance duration, l_r is the traveled length, \dot{b}_m is the maximum base velocity, ψ_m is the maximum tilt angle and $\dot{\psi}_i$ is the angular velocity at the impact instant.

$t = 1.42$ s, when the robot has fully recovered a firm stance on all three wheels, at which point the supervisor can safely transition back to the standard controller presented in [1].

The recovery behavior from a strong push can be seen in Fig. VI, where the robot is gradually pushed during 0.6 s. As a result, the controller presented in this paper is activated at time $t = 0.5$ s, and the mobile base begins accelerating at its maximum capacity. But this is insufficient to recover balance and the robot eventually falls, triggering the fall manager at time $t = 1.6$ s in order to minimize the impact of the body

on the ground. We can observe that in this case, failure is mostly due to the physical limits of the robot.

Finally, in Fig. 8, we compare some results about the performance of the control law. The performance is measured in terms of traveled length by the robot, maximum velocity of the mobile base, maximum tilt angle and impact angular velocity. The disturbance is quantified by its duration and the angular tilt velocity at the end of the disturbance. The table is organized in three push categories, with some variations between the duration of the disturbance and the angular velocity transmitted to the robot.

Concerning the weak pushes, we can note that the robot barely moves because the disturbance does not need a strong effort to recover. Except the second case, the robot is not moving due to the estimated impact velocity limit which is not reached.

About the medium and strong pushes, we can note firstly that the robot needs to move in all cases in order to recover from the disturbance. Also, we can note that the impact angular velocity is bounded at a value equal to the estimated impact velocity limit (in these experiments $0.8rd/s$). This result validates the impact estimator. Finally, we can note that the travelled length is more related to the disturbance duration, whereas the mobile base maximum velocity is more related to the initial angular velocity. This makes sense because in the case of long pushes, the tilt angle grows more, and the robot needs more distance to recover its balance. At the opposite, when the disturbance is strong but short in time, more acceleration and velocity are needed to avoid the fall.

VIII. CONCLUSION

We have presented a new MPC scheme for the robot Pepper, which improves on the controller previously introduced in [1], by introducing a capacity to recover from tilting, when the robot is pushed strongly. It has been validated experimentally in various situations, but there are still situations where the physical limits of the robot do not allow to avoid falling. In such cases, the proposed approach could be improved by using arm motion, to generate an additional angular momentum.

We have currently not taken into account potential rotations around the vertical axis, during both the perturbation and recovery process. In such situations, the behavior of the robot can be much more complex to control since its mobile base is not holonomic anymore, with only two wheels in contact with the ground. This can happen especially in case of a perturbation while the robot is already moving. Preliminary experimental results show a strong degradation of the performance of our controller in such situations.

if the robot detects that obstacles in its surrounding don't allow the recovery process to take place, it currently switches immediately to a safe fall management behavior. This could also be improved.

Here, we have considered only motion of the robot on a planar, horizontal ground. On a ground with constant inclination, the situation is very similar, and the only necessary modification for the proposed controller to work seamlessly

is to shift the position of the CoP on the ground according to the inclination of the ground. Situations where the inclination is not constant are more problematic though and require further refinements.

ACKNOWLEDGMENTS

The research presented in this paper has been partially funded by the ROMEO2 project.

REFERENCES

- [1] J. Lafaye, D. Gouaillier, and P.-B. Wieber, "Closed loop predictive control of an omnidirectional wheeled humanoid robot," pp. 336–341, 2014.
- [2] R. Xiaogang, C. Jing, C. Jianxian, and D. Lizhen, "Balancing control of two-wheeled upstanding robot using adaptive fuzzy control method," in *IEEE International Conference on Intelligent Computing and Intelligent Systems*, 2009, pp. 95–98.
- [3] F. Tao, L. Tao, W. Xu, Z. Xu, and Z. Meng, "Modeling and implementation of two-wheel self-balancing robot equipped with supporting arms," in *IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2011, pp. 713–718.
- [4] B. Mahler and J. Haase, "Mathematical model and control strategy of a two-wheeled self-balancing robot," in *39th Annual Conference of the IEEE Industrial Electronics Society, IECON 2013*, 2013, pp. 4198–4203.
- [5] O. Jamil, M. Jamil, Y. Ayaz, and K. Ahmad, "Modeling, control of a two-wheeled self-balancing robot," in *International Conference on Robotics and Emerging Allied Technologies in Engineering*, 2014, pp. 191–199.
- [6] M. Wada and S. Mori, "Modeling and control of a new type of omnidirectional holonomic vehicle," in *International Workshop on Advanced Motion Control*, 1996, pp. 265–270.
- [7] J. Gohl, R. Rjamani, L. Alexander, and P. Starr, "The development of tilt-controlled narrow ground vehicles," in *American Control Conference*, 2002, pp. 2540–2545.
- [8] H. Furuichi, H. Jian, T. Matsuno, and T. Fukuda, "Dynamic model of three wheeled narrow tilting vehicle and corresponding experiment verification," in *International Conference on Intelligent Robots and Systems*, 2012, pp. 3728–3733.
- [9] —, "Dynamic model of three wheeled narrow tilting vehicle and optimal tilt controller design," in *International Symposium on Micro-NanoMechatronics and Human Science*, 2012, pp. 331–336.
- [10] J. Chiou, L. Chun-Ying, C. Chih-Liang, and C. Chin-Pin, "Tilting motion control in narrow tilting vehicle using double-loop pid controller," in *Asian Control Conference*, 2009, pp. 913–918.
- [11] A. Herdt, N. Perrin, and P.-B. Wieber, "Walking without thinking about it," in *International Conference on Intelligent Robots and Systems*, 2010, pp. 190–195.
- [12] H. Diedam, D. Dimitrov, P. Wieber, K. Nombaur, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *International Conference on Intelligent Robots and Systems*, 2008, pp. 1121–1129.
- [13] W. Jiuguang, "Humanoid push recovery with robust convex synthesis," in *International Conference on Intelligent Robots and Systems*, 2012, pp. 4354–4359.
- [14] A. Yasin, H. Qiang, Y. Zhangguo, X. Qian, and A. Syed, "Stepping to recover: A 3d-lipm based push recovery and fall management scheme for biped robots," in *International Conference on Robotics and Biomimetics*, 2012, pp. 318–323.
- [15] M. Wentao, Q. Guihe, and L. Ju-jang, "Humanoid push recovery strategy for unknown input forces," in *International Conference on Mechatronics and Automation*, 2009, pp. 1904–1909.
- [16] S.-J. Yi, Z. Byoung-Tak, D. Hong, and D. Lee, "Online learning of a full body push recovery controller for omnidirectional walking," in *International Conference on Humanoid Robots*, 2011, pp. 1–6.
- [17] A. Yasin, H. Qiang, X. Qian, and M. Sultan, "Humanoid robot push recovery through foot placement," in *International Conference on Mechatronics and Automation*, 2012, pp. 59–63.
- [18] S. O. Madgwick, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays."