

An Integrated Dynamic Fall Protection and Recovery System for Two-Wheeled Humanoids

Grazia Zambella , Simone Monteleone, Edwin Paul Herrera Alarcón, Francesca Negrello , Gianluca Lentini , Danilo Caporale , Giorgio Grioli , Manolo Garabini , Manuel Giuseppe Catalano , and Antonio Bicchi 

Abstract—Robots face the eventuality of falling. Unplanned events, external disturbances and technical failures may lead a robot to a condition where even an effective dynamic stabilization is not sufficient to maintain the equilibrium. Therefore, it is essential to equip robotic platforms with both active and passive fall protection means to minimize damages, and enable the recovery and restart without physical human intervention. This work introduces a method to design an integrated safety system for two-wheeled humanoids. As a case study, the method is applied to a robot and experimentally tested under several conditions corresponding to different causes of robot instability, such as motor jamming, external disturbances, and sudden shut-down.

Index Terms—Wheeled robots, robot safety, performance evaluation and benchmarking.

I. INTRODUCTION

IN MANY applications where robots are required to move in restricted spaces with small clearance, two-wheeled [1]–[3] or two-legged platforms [4] are often a choice. Their agility and responsiveness is however intrinsically connected to the unstable nature, which requires dynamic controllers to be managed. In the literature many contributions address robot balancing, locomotion, fall detection and recovery, as for example centroid-based control [5], model-predictive control (MPC) [6], zero-moment-point based approaches [7], whole-body control schemes [8], human inspired strategies as in [9]–[11], or capture point based approaches [12].

Manuscript received September 10, 2019; accepted January 19, 2020. Date of publication February 3, 2020; date of current version February 17, 2020. This letter was recommended for publication by Associate Editor Dr. Katsu Yamane and Editor P. Rocco upon evaluation of the reviewers' comments. This work was supported in part by the Horizon 2020 research program EURO-BENCH under Grant 779963 as funded project DYSTURBANCE. (Corresponding author: Grazia Zambella.)

G. Zambella, S. Monteleone, E. P. Herrera Alarcón, D. Caporale, and M. Garabini are with the Centro di ricerca E. Piaggio and the Information Engineering Department, University of Pisa, 56122 Pisa, Italy (e-mail: gr.zambella@gmail.com; simone.monteleone@iit.it; edwinpha@gmail.com; d.caporale@centropiaggio.unipi.it; manolo.garabini@gmail.com).

F. Negrello, G. Grioli, and M. G. Catalano are with the Soft Robotics for Human Cooperation and Rehabilitation, Fondazione Istituto Italiano di Tecnologia, 16163 Genova, Italy (e-mail: francesca.negrello@gmail.com; giorgio.grioli@gmail.com; manuel.catalano@iit.it).

G. Lentini and A. Bicchi are with the Centro di ricerca E. Piaggio and the Information Engineering Department, University of Pisa, 56122, Pisa, Italy, and also with the Soft Robotics for Human Cooperation and Rehabilitation, Fondazione Istituto Italiano di Tecnologia, 16163 Genova, Italy (e-mail: g.lentini88@gmail.com; antonio.bicchi@unipi.it).

This letter has supplementary downloadable material available at <https://ieeexplore.ieee.org>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2020.2970951

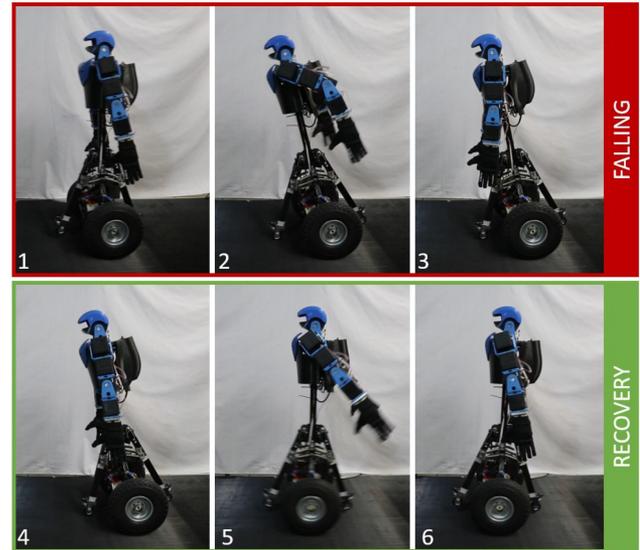


Fig. 1. Photo-sequence of Alter-Ego with the proposed safety system during fall, shut down and recovery.

However, no matter how good the stabilizing controller is, robots face the eventuality of falling (Fig. 1, top row). Falling may be caused by different events: hardware related failures (e.g. motor jamming Fig. 2(a)), power losses (Fig. 2(b)), software problems or communication errors (e.g. the robot can hit an undetected obstacle as in Fig. 2(c)), an unexpected situation (Fig. 2(d)) or an external disturbance that the balance control could not reject (Fig. 2(e)). Therefore, robots operating in real scenarios must be capable of coping with a fall, and such event should be managed in a safe way for the robot and the people around it. Moreover, after a fall the robot should be able to autonomously stand up and recover its operational posture.

To handle a fall, researchers proposed several active strategies for humanoid robots. The most basic approach consists in stopping the robot when a possible fall is predicted [13]. Other strategies minimize the damage caused by a fall by locating the contact point on the backpack or on specific areas that are protected [14], mimicking martial arts techniques [15], [16]. Moreover, it is conceivable to use arms to prevent or absorb the effects of falling [17].

However, a number of robot failure modalities exists that do not allow the control to react at all, and passive systems have been proposed to minimize the effects of a fall. Typical

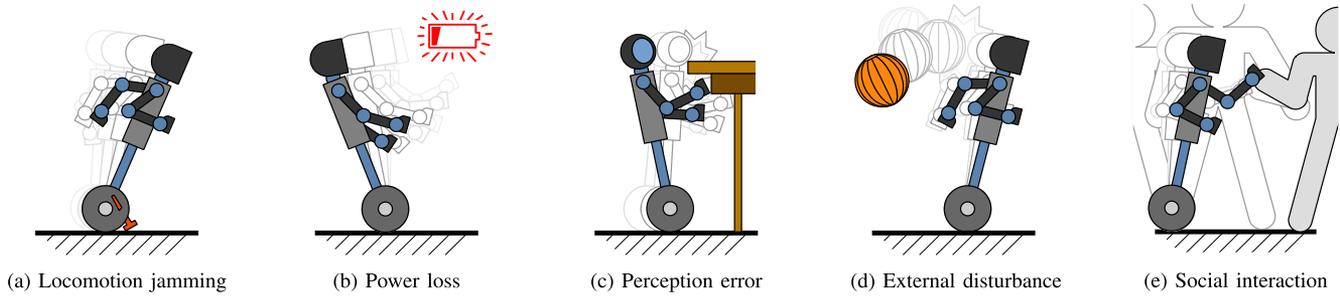


Fig. 2. Examples of robot balance failures due to different causes.

examples are soft covers made of high energy absorption materials or passive overload mechanisms that disengage actuators transmission when impacts occur [18]. It is possible to further reduce the collision forces by introducing compliance in the transmission [19]. However, purely passive systems may alter the robot dynamics, up to the point of invalidating the stabilizer (e.g. by introducing un-modelled oscillations), and do not guarantee system recoverability.

In [20] a comprehensive review of control methods for two wheeled humanoids is presented. From this, it is possible to deduce that the previously discussed methods have a limited applicability to Golem-style robots. One example of safety system for such robots is Cardea's Kickstands,¹ which is composed of two active links. When an error (e.g. over-inclination) is detected, a controller deploys the kickstands to prevent the robot from touching the ground. A similar approach is used also for the Ballbot [21], that has three active legs. In this case, the transition between Static Stable State, in which the robot lies on three legs, and Locomotion State (the legs do not touch the ground) is regulated by an external controller.

We propose a novel integrated dynamic falling protection and recovery system (ISS - Integrated Safety System) to protect the robot from permanent damages, leveraging on its dynamics. This represents a substantial advancement with respect to state of art Golem-style robots, since the ISS allows to protect the robot also in case of power losses while avoiding uncontrolled reactions of the stabilizer. The idea underlying our approach is to deeply integrate a passive (hardware) and active (software) layer, so that, in case of failure, the passive safety layer engages while the control is turned off, reducing the reaction forces generated by the fall.

Lastly, to allow the robot to recover its working posture (and possibly continue the task that it was performing), it is necessary that both the passive and active components of the system are designed to allow the robot to stand up without any external intervention.

This work presents a detailed model of a two-wheeled humanoid robot to provide design guidelines for the passive safety layer. The hybrid safety system concept and its application to a case study are presented. Finally, extensive experimental tests are performed to validate the system.

¹<http://www.ai.mit.edu/projects/cardea/technical/kickstands.shtml>

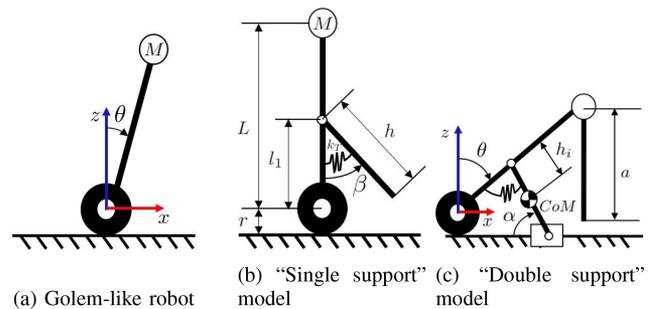


Fig. 3. Simplified balancing dynamics of a Golem-like robot (a) and Schematic of the proposed concept (b)–(c): main dimensions and parameters. Only one kickstand is shown for readability.

II. INTEGRATED SAFETY SYSTEM CONCEPT

The dynamics of a Golem-style robot [22] (see Fig. 3(a) for the definition of the various symbols) can be approximated as

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = H\tau, \quad (1)$$

where

$$q = \begin{bmatrix} x \\ \theta \end{bmatrix} \text{ and } H = \begin{bmatrix} 1/r \\ -1 \end{bmatrix}$$

are the robot state and transmission matrix, respectively, B is the robot inertia matrix, C is the Coriolis and centrifugal forces matrix, G is the gravity vector, while x and θ are, respectively, the horizontal position and the pitch angle of the robot.

System (1) is under-actuated, because the rank of H is one, while the dimension of q is two. Since the equilibrium in $q_0 = [0, 0]^T$ is unstable, it must be actively stabilized. Although, many techniques are available in the literature, because of practical limits (coming e.g. from maximum motor power, controller and sensor bandwidth, ground friction) the equilibrium has a finite Region of Asymptotic Stability (RAS).

Any disturbance large enough to move the state of the robot out of its RAS, will ultimately result in a failure of balancing.

A. Proposed Approach

To manage all conditions that bring the robot state outside its RAS, it is possible to integrate a passive kickstand protection system, as shown in Fig. 3(b).

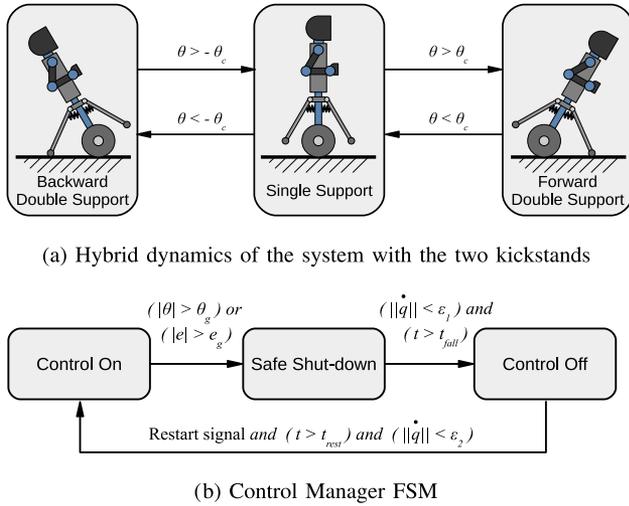


Fig. 4. The two hybrid systems describing the integrated fall protection and recovery system. Variable e defines the error threshold over linear position and ϵ_g are the error thresholds on the $\|\dot{q}\|$.

Robot dynamics becomes then a hybrid system with one “single support” phase (the default working condition of the robot) characterized by a dynamics similar to (1), and a “double support” phase with the robot leaning on a kickstand. As a matter of fact, practical implementations of the idea may require two kickstands, fore and back, hence two distinguished double support phases (see Fig. 4(a)). The dynamics of the robot in double support phase will be derived in detail in Sec. III.

Transition from one condition to the other is essentially geometric and can be described by the Finite State Machine (FSM) in Fig. 4(a). The idea is that the two double support states have one equilibrium each (θ_{eq+} and θ_{eq-}) that, given proper tuning of the kickstands design parameters (see Sec. III), will (i) be stable, and (ii) enable the robot to recover and go back to single support.

To achieve this, we implement a control manager with the FSM shown in Fig. 4(b), which regulates the activation and deactivation of the control system to avoid instability of the closed loop hybrid system.

Fig. 5 describes all possible combinations between the discrete states of the robot and the controller. We expect the hybrid system to evolve following the black arrows. During normal operation the robot is in single support with the control on (left top box in Fig. 5). If a failure condition occurs, the control manager activates the safe shut-down. The system evolution causes the robot transition towards a double support state. Once the system reaches the double support equilibrium, the controller is switched off (bottom right box). After a proper time, if the equilibrium is maintained, the manager reboots the control system (top right box) and the robot recovers the operative condition (left top box).

Note that, the dotted arrows represent system state transitions that are possible but unlikely to happen. In detail, the blue connection does not occur as long as we chose $\theta_g < \theta_c$, where θ_g is the guard condition to activate the safety mode and θ_c is the pitch angle when the kickstand contacts the ground. The green transitions usually do not happen as they assume that the robot

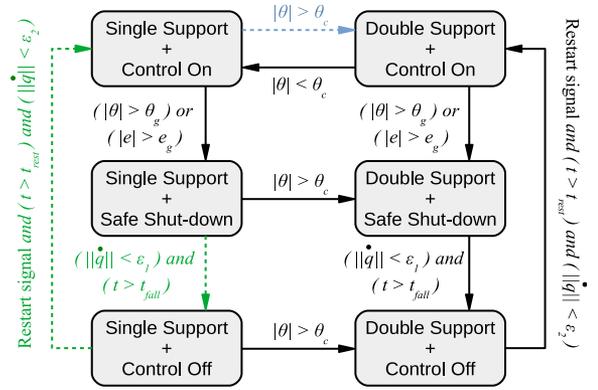


Fig. 5. Full FSM describing the combined discrete behavior of the system physics and the control manager state. Note that the two double support conditions shown in Fig. 4 are grouped in one for readability.

is able to maintain the equilibrium when the control is off for a time longer than $t_{fall} + t_{rest}$, where t_{fall} is the duration of the fall, while t_{rest} is the time to wait before restarting the system. To this aim, a properly sized timer regulates the transition among states and the restart of the controller. In the unlikely eventuality that at $t > t_{fall}$ the system is still in equilibrium on the single support with the control off, the controller can be safely switched on because the starting condition is inside the RAS.

In the following section we discuss how to compute the design parameters of the ISS, taking into account the physical constraints that guarantee the correct functionality of the robot.

III. PASSIVE SAFETY SYSTEM MODELLING

As discussed above, the ISS has two fundamental functions: (i) regulate the transition from the single to the double support state in a safe manner for the robot and the people surrounding, (ii) guarantee the recovery of the single support state, e.g. by ensuring the double support equilibrium is inside the RAS of the balance control system. These requirements introduce the following constraints to the ISS passive layer design.

1) *Torque Limits Constraint*: the motor torque is used to control both the horizontal position x and pitch angle θ . Therefore, the inertia of the wheel, of the base and the body influence the maximum recovery angle θ_r . Subscript r indicates the variables related to the recovery task. Starting from (1) and imposing initial speeds $\dot{\theta}_0 = 0$ and $\dot{x}_0 = 0$, we get that

$$\ddot{q}_r = B(q_r)^{-1} (H\tau - G(q_r))$$

with τ the torque computed by the control law to stabilize the system in the equilibrium q_0 , which can be recovered only if $\dot{\theta}_r < 0$ and $\ddot{x}_r > 0$. The limit conditions beyond which the upward position can not be recovered can be solved equating

$$B(q_r)^{-1} G(q_r) = B(q_r)^{-1} H\tau. \quad (2)$$

Inverting (2), we can express θ_r as a function of τ . The most important constraint limiting the maximum angle from which the equilibrium can be recovered, is the maximum motor torque (τ_{max}). Substituting $\tau = \tau_{max}$ in (2), we can express θ_r as a function of τ_{max} . Therefore, computing θ_r following these considerations, the equilibrium angles for double support states

(θ_{eq}) must satisfy

$$|\theta_{eq}| \leq \theta_r . \quad (3)$$

Note that an additional constraint can be imposed on the maximum allowable translation of the robot during recovery (x_{max}), to avoid further accidental collisions with the environment.

2) Geometric Constraints:

a) *Arms to ground contact:* it should be avoided that during the fall any part of the robot hits the ground. As shown in Fig. 3(c), the hands are the most exposed part. This poses a constraint on the maximum pitch angle reached during the fall (θ_{max}) as

$$\theta_c < \theta_{max} = \arccos\left(\frac{a-r}{L}\right) \quad (4)$$

where a is the arm length, L is the distance between the center of the wheels and the shoulders, and r is the radius of the wheels. Subscript c indicates the variables at the contact time.

b) *Passive opening condition:* to ensure that the kickstand opens when touching the ground, we must impose that at the contact $\alpha_c \leq \frac{\pi}{2}$, which leads to

$$r + l_1 c_{\theta_c} < h_{min} .$$

From now on, we use the convention that s_θ and c_θ are respectively the $\sin(\theta)$ and $\cos(\theta)$ functions. Additionally, to avoid kickstand jamming during robot operations, the spring preload angle (β) must be lower than

$$\beta_c = \frac{\pi}{2} - \alpha_c .$$

c) *Inclined or irregular terrain:* considering a Golem-style robot crossing an inclined surfaces, the maximum slope that the system can climb imposes a geometrical constraint to the kickstand design, which should not get in contact with the ground. Defined M_{eq} as the overall mass including the wheels, the maximum climbing angle (γ_{max}) should be evaluated as function of the maximum allowable torque

$$\tau = M_{eq} g r s_\gamma \rightarrow \tau_{max} = M_{eq} g r s_{\gamma_{max}}$$

$$\gamma_{max} = \arcsin\left(\frac{\tau_{max}}{M_{eq} g r}\right) .$$

Then, the geometric maximum slope α_s can be computed by

$$r \begin{bmatrix} s_\gamma \\ c_\gamma \end{bmatrix} + l_1 \begin{bmatrix} -s_{\theta_c} \\ c_{\theta_c} \end{bmatrix} - h \begin{bmatrix} s_{(\beta_d - \theta_c)} \\ c_{(\beta_d - \theta_c)} \end{bmatrix} = d \begin{bmatrix} -c_\gamma \\ s_\gamma \end{bmatrix}$$

where β_d is the kickstand angle when the system is in standing position. Imposing that $\gamma < \gamma_{max}$ provides an additional constraint to h , l_1 and β_d (Fig. 6(a)).

Lastly, in case of terrain irregularities or obstacles to overcome, it is conservative to impose that the distance from the kickstand tip to the ground is larger than r . A more detailed analysis of the problem of ascending steps with two wheeled robots is reported e.g. in [23].

3) *Dynamic Constraints:* we model system dynamics to choose the spring constant, the length of the kickstand, the position of the joint and the spring preload. We considered the

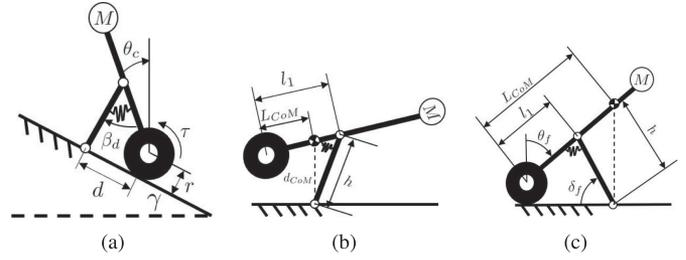


Fig. 6. (a) Slope limit condition. (b) and (c) Tipping limit condition for Case 1 ($l_1 > L_{CoM}$) and Case 2 ($l_1 < L_{CoM}$).

free fall dynamics and the constrained one. The first provides the initial condition to the latter. Constrained dynamics models the kickstand as a prismatic joint that can slide along the x axis, and defines system overshoot (Fig. 3(c)). Here we impose the condition that hands do not touch the ground during the fall.

a) *Free fall dynamics:* the angular velocity at which the kickstand contacts the ground represents the initial condition of the constrained dynamics and is evaluated as

$$\frac{1}{2} M_{eq} V_{0_{CoM}}^2 + MgL = MgL c_{\theta_c} + \frac{1}{2} I_{eq} \dot{\theta}_c^2$$

where I_{eq} is the inertia of the system rotating around the wheels axis, and $V_{0_{CoM}}$ is the initial speed of the CoM.

b) *Constrained dynamics:* starting from the contact point θ_c , with initial speed $\dot{\theta}_c$, we write the dynamics assuming that during the fall, wheels are not moving, so that their axis becomes a pivot around which the robot rotates. Due to the prismatic joint, the following constraint holds

$$r + l_1 \cos(\theta) - h \sin(\alpha) = 0 \quad (5)$$

$$-l_1 \dot{\theta} \sin(\theta) - \dot{\alpha} h \sin(\alpha) = 0$$

Dynamics has the form

$$B(p)\ddot{p} + C(p, \dot{p})\dot{p} + G(p) = 0 , \text{ with } p = \begin{bmatrix} \theta \\ \alpha \end{bmatrix}$$

with $A\ddot{p} + \dot{A}\dot{p} = 0$ constraint in Pfaffian form.

In $G(p)$, we assume to have only a torsional spring at the joint of the kickstand, and a damping action given by the friction of the real spring during task execution. We compute the dynamics in its standard form

$$\dot{x} = f(x) = \begin{bmatrix} \dot{p} \\ -B(p)^{-1} (C(p, \dot{p})\dot{p} + G(p)) \end{bmatrix}$$

with

$$x = \begin{bmatrix} p \\ \dot{p} \end{bmatrix}$$

which initial conditions are

$$x_0 = \begin{bmatrix} \theta_c & \alpha_c & \dot{\theta}_c & \dot{\alpha}_c \end{bmatrix}$$

$$x_{eq} = \begin{bmatrix} \theta_{eq} & \alpha_{eq} & 0 & 0 \end{bmatrix} .$$

Note that, imposing a desired equilibrium point, an additional constraint (resulting from (3)) on the design parameters is added to the system, given by the static equations

$$G(p_{eq}) = 0.$$

Let $\tilde{x} = x - x_{eq}$, by linearization we have

$$\dot{\tilde{x}} = \left. \frac{\partial f(\tilde{x})}{\partial x} \right|_{x=x_{eq}} = F\tilde{x}$$

where

$$F = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ k_1^\theta & k_2^\theta & 0 & 0 \\ k_1^\alpha & k_2^\alpha & 0 & 0 \end{bmatrix}, \quad \text{with} \quad \begin{aligned} \lambda_{1,2} &= -\sigma_1 \pm \omega_1 i \\ \lambda_{3,4} &= -\sigma_2 \pm \omega_2 i \end{aligned}$$

$\lambda_{1,2}$ and $\lambda_{3,4}$ are the eigenvalues of F , and they have negative real parts, hence the system is stable. The resulting dynamics is $x(t) = x_{eq} + e^{Ft}(x_0 - x_{eq})$. Studying $\theta(t)$ it is possible to extract the maximum overshoot.

c) *Jamming and tipping on the kickstand*: if the ground presents some irregularities, the kickstand may jam and robot may tip around the kickstand. Such condition is avoided by imposing a limit on the robot maximum speed, which depends on the kickstand joint location with respect to the center of mass (CoM).

Case 1. CoM below the kickstand joint ($l_1 > L_{CoM}$): After the jamming, the robot inertia will tend to close the kickstand against its end-stop. Therefore, the whole system will pivot around the contact point as a rigid body (Fig. 6(b)). The tipping speed V_{tip} is given by

$$\frac{1}{2}M_{eq}V_{tip}^2 + M_{eq}g(r + L_{CoM}) = M_{eq}gd_{CoM}$$

where d_{CoM} is the distance between the tip of the kickstand and the CoM. Computing them using design parameters, we obtain

$$d_{CoM} = \sqrt{(l_1 - L_{CoM})^2 + h^2} - 2(l_1 - L_{CoM})h s_{(\theta_c - \alpha_c)}$$

$$V_{tip} = \sqrt{2g(d_{CoM} - r - L_{CoM})}.$$

Case 2. CoM above the kickstand joint ($l_1 < L_{CoM}$): In this case, the robot inertia causes the opening of the kickstand, so that we have two degrees of freedom: δ as the angle between the kickstand and the ground, and θ . The stability limit is when the center of mass is aligned with the kickstand tip (Fig. 6(c)). Considering the mass and inertia of the kickstand negligible with respect to the rest of the system, $\dot{\delta}$ and $\dot{\theta}$ equal to zero, the following relation holds

$$\frac{1}{2}M_{eq}V_j^2 + M_{eq}g(r + L_{CoM}) = \frac{1}{2}K_T \left(\theta_f - \delta_f + \frac{\pi}{2} - \beta_0 \right)^2 + M_{eq}g [h s_{\delta_f} + (L_{CoM} - l_1) c_{\theta_f}]$$

$$h c_{\delta_f} = (L_{CoM} - l_1) s_{\theta_f}.$$

We solved the system iteratively varying θ_f . The tipping speed is equal to the minimum speed that causes the system to rotate

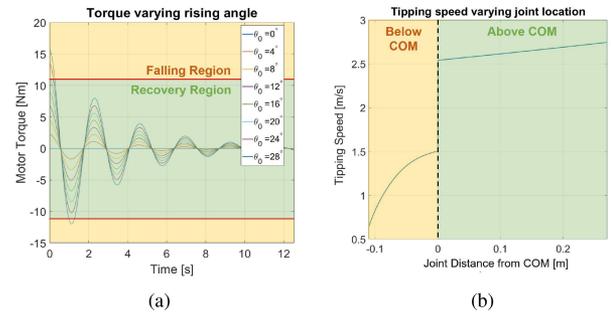


Fig. 7. (a) Simulated required τ starting from different θ_i . (b) Simulated max system speed from jamming and tipping condition.

up to the limit condition:

$$V_{tip} = \min_{\theta_f} (V_j).$$

Intuitively, case 1 max. speed is larger than case 2. Therefore, it is suggested to impose by design the case 1 condition.

IV. IMPLEMENTATION OF THE ISS FOR ALTER-EGO

In this section we present the design of the hybrid fall protection and recovery system for the Alter-Ego robot, a two-wheeled humanoid, introduced in [24].

A. Mechanical Design

The design parameters of the kickstand (see Fig. 3(c)) are the position of the kickstand joint (l_1), its length (h), the spring stiffness (k_T), the preload (β_c), and the maximum elongation of the spring. To select them, the Alter-Ego dynamics has been modeled according to the equations shown in Sec. III. Fig. 7(a) shows the torque required at the wheel axis (τ), as function of the time for different initial pitch angles θ_0 . Since the Alter-Ego platform has two different controllers [24], an LQR one and a whole-body one, which can be used alternatively, we compute τ with the LQR controller which is the most conservative choice. Considering that the maximum continuous torque of the Alter-Ego wheels is 11 Nm, we obtain the recovery region highlighted in green in Fig. 7(a). At the maximum torque corresponds the maximum angle $\theta_i = 17^\circ$ within which it is possible to recover from a fall. The position of the kickstand joint $l_1 = 0.3$ m has been chosen the highest possible compatibly with the other components of the torso, and in any case above the CoM, to reduce the risk of tipping.

Given θ_i and l_1 , the equilibrium condition provides a relationship between k_T and h . From (5) imposing $\alpha = \alpha_c$, h is 0.4 m. Therefore, to guarantee that a free fall from the upright position produces an overshoot not larger than θ_{max} (4), k_T is 11.3 Nm/rad. To avoid unwanted oscillations that may destabilize the system, it is important that the legs do not move until in contact with the ground. For this reason, the springs have a preload of 30° . Moreover, the maximum spring extension required is about 65° .

Unfortunately, off-the-shelf torsional springs with such stiffness have low extension range, and do not allow to meet the requirements. Therefore, we combined both torsional and linear

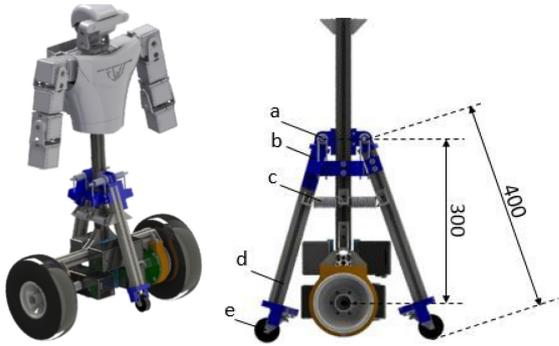


Fig. 8. CAD model of Alter-Ego with passive safety system (dimensions are in mm).

springs to obtain a stiffness equivalent to the desired one while matching the elongation requirement.

Fig. 8 shows the passive mechanism symmetrically placed on the front and on the back of the robot. The passive mechanism consists of a pin joint (a), two torsional springs (b) with $k_{tr} = 3.2$ Nm/rad, maximum angle 85° , four linear springs (c) with $k_{ln} = 320$ N/m with maximum extension 175 mm, a leg (d) and a caster wheel (e) at the leg's tip to emulate the linear sliding constraints in Fig. 3(c). k_T can be computed as

$$k_T = 2k_{tr} + 4k_{ln}l_p^2.$$

The two torsional springs are located at the joint axis while the linear ones are at $l_p = 60$ mm from the joint. This choice allowed to have a symmetric load on the joint.

Lastly, Fig. 7(b) shows the tipping speed as function of the joint placement with respect to the CoM ($L_{CoM} = 0.2$ m), with the current design, the maximum speed the robot can safely operate is about 2.6 m/s.

B. Safety Mode Algorithm

As introduced in the previous section, Alter-Ego can be controlled with two approaches: a traditional LQR based controller [24] and a dynamic whole-body one [25]. The latter couples arm and pitch dynamics such that non-zero torques on the arms are commanded to compensate a pitch error (even if the errors on the shoulder joints position and velocity are zero) or the effect of an external disturbance. This control approach allows to reject disturbances of higher amplitude and also to recover in a shorter time and space if compared to conventional approaches in which the balancing relies only on the mobile base.

A manager node to detect the falling condition is implemented in ROS (Robot Operating System). At each time step, it checks if the critical condition on the pitch angle of Alter-Ego ($|\theta| > \theta_c$) is reached or if a human operator triggers a stop. If one of these situations occurs, the Safety Mode is activated. The safety mode sends zero PWM (Pulse Width Modulation) commands to the base actuators forcing the motors to freeze and, once this happens, it turns them off. When the recovery task can be executed, the manager resets the system, turns on the motors and re-activates the control node. Note that, the safety mode works at a higher layer and independently from the selected stabilizing controller.

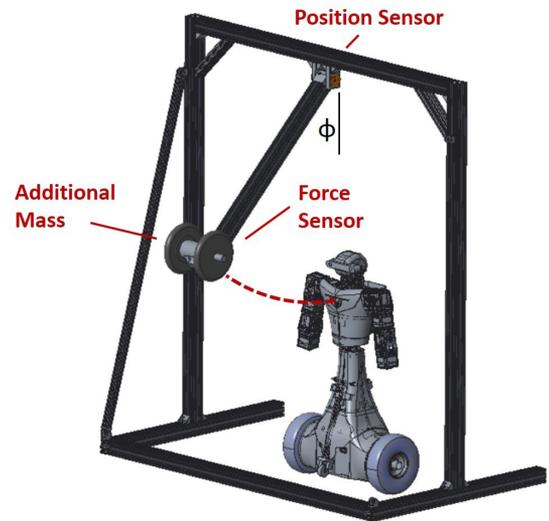


Fig. 9. CAD model of the experimental test-bench.

V. EXPERIMENTAL VALIDATION

To validate the ISS, it was tested under different fall conditions. Experiments are organized in three categories: robot internal failure, external disturbance, and recovery. The two failure categories are evaluated with both zero linear velocity and at the maximum operative speed of the robot. In particular, the first category recreates the falling conditions due to power loss and locomotion jamming. The falling conditions due to external impulsive disturbance and perception error correspond to the second category. The recovery category reports the experiments where the robots autonomously recover its operative position.

A. Set-Up

The experiments were executed on an Alter-Ego robot equipped with the ISS. During the experiments we recorded the robot internal measures such as robot pitch angle, motor positions and torques. The motion of the passive leg (β) is extracted by videos (Kinovea).

To recreate the robot internal failure condition, two modalities were implemented, the power loss modality, which forces the motors shut down, and the motors block modality, which short circuits the DC motors on the wheels. In the latter case, the robot falls while the arms controller is still active. For these experiments, we used the whole-body controller. Therefore, the arms will move in the opposite direction trying to compensate the pitch angle. Exceeding the safety boundaries triggers the Safety Mode.

To generate reproducible impulsive loads we used the test-bench presented in [26], (Fig. 9).

Impulsive load test are executed increasing the pendulum initial height, up to the impulsive force that destabilizes the robot triggering the passive safety system. During impulsive load with linear velocity the robot is started in front of the fixed pendulum and is commanded to move at its maximum speed against the structure.

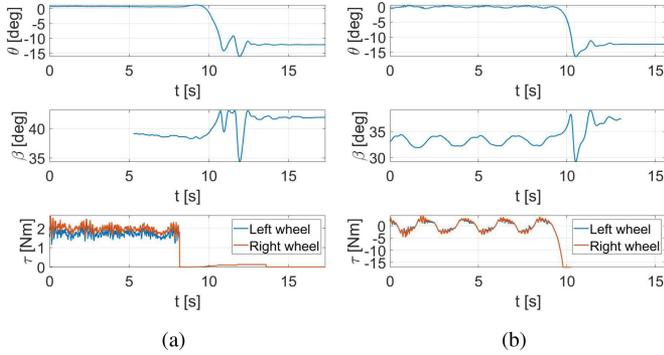


Fig. 10. Internal fault results for (a) motor block and (b) full shut down with zero linear velocity.

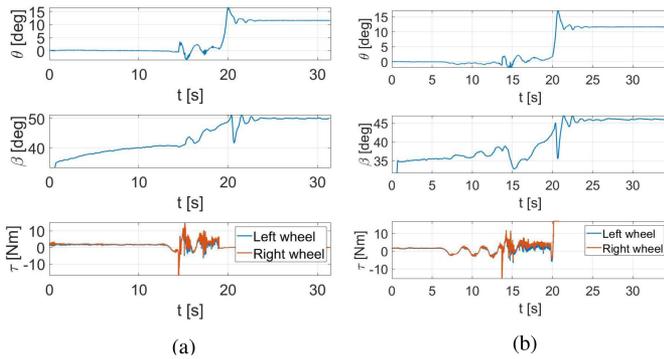


Fig. 11. Internal fault results for (a) motor block and (b) full shut down at robot maximum linear velocity (0.25 m/s).

B. Results

Robot Internal Failure: Fig. 10, shows the robot pitch angle (θ), and motor torque for both motor block and full shut down conditions with zero linear velocity. The fault modality is triggered around $t = 9$ s and the pitch angle reaches the θ_{eq} in about 2 s. The larger β_{max} is measured for the motor block modality, while in both experiments the measured β is noisy due to the data extraction procedure. Concerning the torque measurements, it is worth noting that, when the shut down is triggered, the measurement is lost, while during the motor block τ goes to zero and is maintained. Fig. 11 reports the same experiments executed with a linear velocity equal to 0.25 m/s (Alter-Ego maximum operative speed). The values of θ , β and τ show the same behaviours shown in the previous experiment.

External Disturbance: Fig. 12 shows θ , β and the absorbed energy for the external load case. Above an impulsive force of 500 N (pendulum initial angle about 20°), the stabilizer fails and the safety system activates. Accordingly, θ , β and the absorbed energy increase with the magnitude of the disturbance. It is worth noting that the larger pitch variance is at 700 N, because the controller overreacts, generating a higher maximum pitch. Fig. 13 shows the data of the impacts with maximum linear velocity. The photo-sequence above the graphs shows the different phases of the experiment. After the impact, the control system tries to stabilize the robot by rising the arms. Once the fall is detected, the safety mode is activated, driving the robot to the

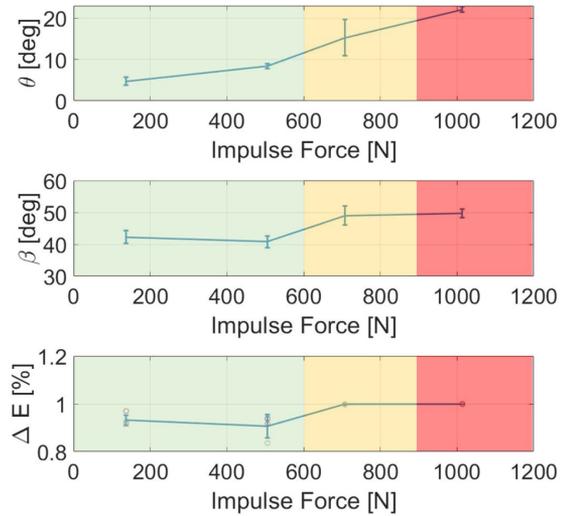


Fig. 12. Pitch, beta and energy variance vs impulsive force. Green box shows the range of impulse forces that the active control is capable to handle, the yellow box shows the range in which the control is active but over-reacts, while the red zone is the range at which the impulse activates the safety mode.

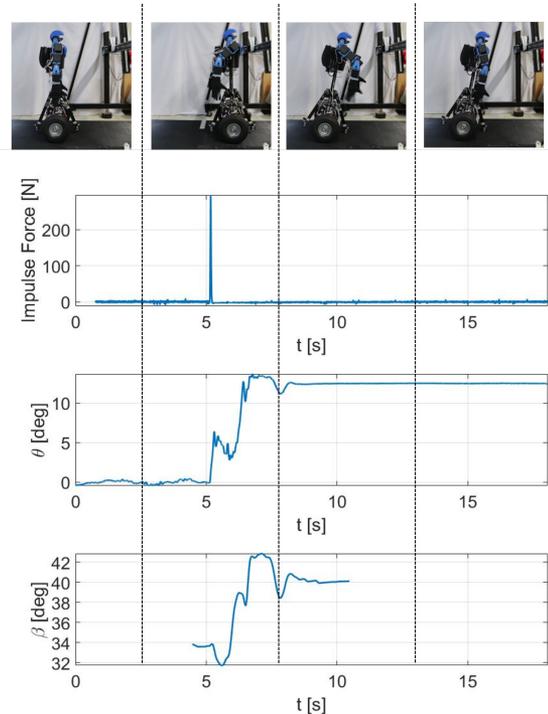


Fig. 13. Case $\dot{x} \neq 0$ impulsive force, pitch, beta in the time.

homing position. Comparing this test with Fig. 12, we highlight that increasing the linear velocity, it is required a lower force to destabilize the system because less motor torque is available for the balance control.

Recovery: In this set of experiments, the system starts from the double support state condition presented in the Sec. II-(a). Then, the system is turned on and recovers the single support state. Fig. 14 shows that the center of mass location determines

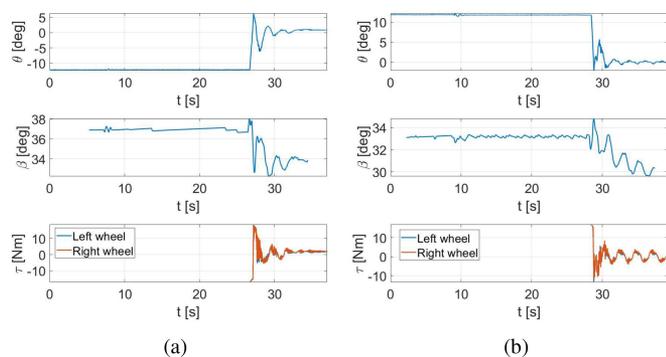


Fig. 14. Recovery from backward (a) and from forward fall (b).

different load requirements for the forward and backward recovery torque. Accordingly, τ for a forward recovery requires less effort than the backward case because the center mass is slightly displaced toward the rear.

VI. CONCLUSION

In this work we presented an integrated fall protection and recovery system for two-wheeled humanoids. A method to design the passive and active components, and their integration is discussed. The ISS was validated under different fall conditions, considering both internal failure and external disturbances, that recreate the possible situation of fall in everyday life. Results show that the ISS does not limit the performance of the stabilizer and that it effectively controls robot falls. Moreover, it allows the robot to recover the operative state without human intervention, independently on the falling direction.

ACKNOWLEDGMENT

The content of this publication is the sole responsibility of the authors. The European Commission or its services cannot be held responsible for any use that may be made of the information it contains.

REFERENCES

- [1] K. D. Mullens *et al.*, "Segway robotic mobility platform," in *Mobile Robots XVII*, vol. 5609, International Society for Optics and Photonics, 2004, pp. 207–220.
- [2] M. Stilman, J. Olson, and W. Gloss, "Golem krang: Dynamically stable humanoid robot for mobile manipulation," in *Proc. Int. Conf. Robot. Autom.*, 2010, pp. 3304–3309.
- [3] D. Ruiken, J. Cummings, U. Savaria, F. Sup IV, and R. Grupen, "ubot-7: A dynamically balancing mobile manipulator with series elastic actuators," in *Proc. IEEE-RAS 17th Int. Conf. Humanoid Robot.*, Birmingham, 2017, pp. 676–682.
- [4] J. Engelsberger *et al.*, "Overview of the torque-controlled humanoid robot toro," in *Humanoids*, Nov. 2014, pp. 916–923.
- [5] B. Henze, A. Dietrich, M. A. Roa, and C. Ott, "Multi-contact balancing of humanoid robots in confined spaces: Utilizing knee contacts," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2017, pp. 697–704.
- [6] A. Herdt, N. Perrin, and P. Wieber, "Walking without thinking about it," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 190–195.
- [7] M. Vukobratović and J. Stepanenko, "On the stability of anthropomorphic systems," *Math. Biosciences*, vol. 15, no. 1–2, pp. 1–37, 1972.
- [8] M. Zafar and H. I. Christensen, "Whole body control of a wheeled inverted pendulum humanoid," in *Humanoids*, 2016, pp. 89–94.
- [9] M. Abdallah and A. Goswami, "A biomechanically motivated two-phase strategy for biped upright balance control," in *Proc. Int. Conf. Robot. Autom.*, 2005, pp. 1996–2001.
- [10] C. G. Atkeson and B. Stephens, "Multiple balance strategies from one optimization criterion," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2007, pp. 57–64.
- [11] K. Yamamoto, "Control strategy switching for humanoid robots based on maximal output admissible set," *Robot. Auton. Syst.*, vol. 81, pp. 17–32, 2016.
- [12] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *Humanoids*, pp. 200–207.
- [13] C. G. Atkeson *et al.*, "No falls, no resets: Reliable humanoid behavior in the darpa robotics challenge," in *Humanoids*, 2015, pp. 623–630.
- [14] S. Lee and A. Goswami, "Fall on backpack: Damage minimization of humanoid robots by falling on targeted body segments," *J. Comput. Nonlinear Dyn.*, vol. 8, no. 2, 2013, Art. no. 021005.
- [15] K. Fujiwara *et al.*, "The first human-size humanoid that can fall over safely and stand-up again," in *Proc. Int. Conf. Intell. Robots Syst.*, 2003, vol. 2, pp. 1920–1926.
- [16] K. Ogata, K. Terada, and Y. Kuniyoshi, "Falling motion control for humanoid robots while walking," in *Humanoids*, 2007, pp. 306–311.
- [17] J. Lee, W. Choi, D. Kanoulas, R. Subburaman, D. G. Caldwell, and N. G. Tsagarakis, "An active compliant impact protection system for humanoids: Application to walk-man hands," in *Humanoids*, 2016, pp. 778–785.
- [18] K. Kobayashi, T. Yoshikai, and M. Inaba, "Development of humanoid with distributed soft flesh and shock-resistive joint mechanism for self-protective behaviors in impact from falling down," in *Proc. Int. Conf. Robot. Biomimetics*, 2011, pp. 2390–2396.
- [19] F. Negrello *et al.*, "A modular compliant actuator for emerging high performance and fall-resilient humanoids," in *Humanoids*, 2015, pp. 414–420.
- [20] R. P. M. Chan, K. A. Stol, and C. R. Halkyard, "Review of modelling and control of two-wheeled robots," *Annu. Rev. Control*, vol. 37, no. 1, pp. 89–103, 2013.
- [21] U. Nagarajan, A. Mampetta, G. A. Kantor, and R. L. Hollis, "State transition, balancing, station keeping, and yaw control for a dynamically stable single spherical wheel mobile robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 998–1003.
- [22] C. Acar and T. Murakami, "Multi-task control for dynamically balanced two-wheeled mobile manipulator through task-priority," in *Proc. IEEE Int. Symp. Ind. Electron.*, 2011, pp. 2195–2200.
- [23] T. S. Chen, P. Hazelwood, and K. A. Stol, "Step ascent modelling of a two-wheeled robot," *Int. Proc. Conf. Mechatronics Mach. Vision Pract.*, 2012, pp. 310–315.
- [24] G. Lentini *et al.*, "Alter-ego: A mobile robot with functionally anthropomorphic upper body designed for physical interaction," *IEEE Robot. Autom. Mag.*, vol. 26, no. 4, pp. 94–107, 2019.
- [25] G. Zambella *et al.*, "Dynamic whole-body control of unstable wheeled humanoid robots," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3489–3496, Oct. 2019.
- [26] F. Negrello *et al.*, "Benchmarking hand and grasp resilience to dynamic loads," in *IEEE Robot. Autom. Lett.*, p. 1, 2020, doi: 10.1109/LRA.2020.2969180