HAL
open science

Gyrubot: nonanthropomorphic stabilization for a biped

Nikita Mikhalkov, Alexey Prutskiy, Semyon Sechenev, Dmitry Kazakov, Alexey Simulin, Dmitry Sokolov, Igor Ryadchikov

To cite this version:

Nikita Mikhalkov, Alexey Prutskiy, Semyon Sechenev, Dmitry Kazakov, Alexey Simulin, et al.. Gyrubot: nonanthropomorphic stabilization for a biped. ICRA 2021 - IEEE International Conference on Robotics and Automation, May 2021, Xi'an / Virtual, China. hal-03172031

HAL Id: hal-03172031

https://hal.science/hal-03172031v1

Submitted on 17 Mar 2021

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.

Gyrubot: nonanthropomorphic stabilization for a biped

Nikita Mikhalkov[1], Alexey Prutskiy[2], Semyon Sechenev[1], Dmitry Kazakov[1], Alexey Simulin[1], Dmitry Sokolov[3,4], Igor Ryadchikov[2]

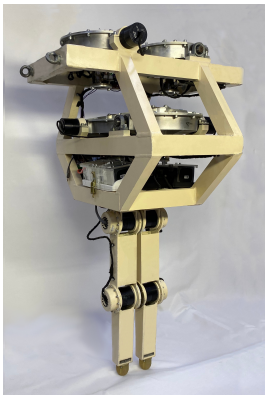
***Abstract*— Demands on leg degrees of freedom and control precision for bipedal robotics are steadily increasing, especially for the tasks involving walking on a rough terrain. In this paper we present an alternative, as well as a working proof-of-concept. Meet *gyrubot*: a 5-link almost planar bipedal robot with a torso complemented by a nonanthropomorphic stabilization system, capable of blindly walking through uneven areas. Despite being almost planar, the robot does not need any support in the frontal plane! This paper describes the mechanical design and the architecture of the controllers. We also provide the experimental evidence of the ability of *gyrubot* to navigate across non-flat terrains.**

I. Introduction

Gyrubot is a bipedal robot: wheels are perfect for a paved road, but if you want a robot to cross a uneven terrain, legs are really the best option. While dynamic bipedal stabilization remains a challenging topic and a relevant benchmark for control design and a multitude of technological processes, a fulgurant progress in this area does not cease to amaze the public. Robots like *Atlas* [1], *Cassie* [2], *Digit* [3] have almost become commercially available and video clips of their capabilities attract hundreds of millions of viewers. They are able to perform somersaults [4] and ride hovershoes [5]. Amazing grace and agility of these robots, up until now available only for humans and animals, rely on the outstanding mechanical design, finest motor control and real-time processing of accurate sensor data.

Unfortunately, legs are difficult to control. Recall that legs have multiple articulations chained together; each one comes with its own mechanical play, the inaccuracies are cumulative. This can rapidly result in a system that is difficult to control, or even unusable. The problem becomes more acute if the robot manufacturing costs are to be reduced. Lower cost generally means more weight, less torque and a great deal of accidental backlash in the mechanical transmission. Here is the immutable truth: the less you pay for the robot parts, the more challenging the control task becomes.

There is, however, a way to enhance the robot's stability even in a presence of hard-to-model factors like backlashes, frictions and mechanical deformations. We can design auxiliary devices capable to improve the stability while performing complex tasks. There is a rich universe for nonanthropomorphic stabilization helpers. Thus, *Handle* [6] and *Ascento* [7] use wheels as a replacement for actuated feet to provide more stability. In [8] augmentation of *ERNIE* robot with a reaction wheel coupled with hybrid zero-dynamics [9] has improved some of the walking parameters of the robot. Another example is *Anywalker* [10], [11], a robot that uses flywheels to provide increased stability of the zero moment point controller. Our work is inspired by [12], we propose to give robots an additional "support point" without touching the rest of the mechanical system. We have created an electromechanical auxiliary stabilization system able to compensate for external disturbances and control errors with aid of control moment gyroscopes (CMGs).



parameter	value
robot height, m	1.18
total mass, kg	39.70
femur length, m	0.25
femur mass, kg	1.59
tibia length, m	0.25
tibia mass, kg	0.75
torso mass, kg	14.90
flywheel mass, kg	2.01
flywheel axial rotational inertia, $\mathrm{kg\,m^2}$	0.01
CMG mass, kg	4.99
flywheel speed, RPM	6000

Fig. 1: Current prototype of the *gyrubot* robot. It has two 2 DoF legs and two scissored pair control moment gyroscopes. The table to the right provides its main mechanical parameters.

A CMG is an electromechanical device consisting of a rapidly rotating flywheel, and actuated gimbals that alter the plane of rotation of the flywheel. When the gimbal actuators produce the control torque, CMG applies a torque in the direction orthogonal to the control axes. The ratio resulting torque / control torque is large enough to provide a wide variety of applications for these devices. They are commonly used for attitude steering of satellites [13] and underwater vessels [14], [15] due to their relatively low power consumption. Also, they have similar applications for motorcycle and bicycle balancing [16], [17], exoskeletons [18] and wheel-based robots [19].

There exist several variations of CMGs, but in this paper we restrict ourselves to constant speed, single gimbal CMGs. *Constant speed* means that the flywheel has a motor that keeps its spinning at a constant velocity. *Single gimbal* means that it has only one actuated gimbal. To create a unidirectional torque, it is possible to assemble two single gimbal CMGs in a scissored pair, where the actuators and flywheels rotate in opposite directions. We refer to such a

[1] Neurolab LTD, Moscow, Russia
[2] Kuban State University, Krasnodar, Russia
[3] Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
[4] Corresponding author: dmitry.sokolov@univ-lorraine.fr

device as to a *scissored pair CMG*, or SP-CMG.

In this paper we present *gyrubot* (refer to Fig. 1), a biped with 2 DoF legs that is stabilized by two orthogonal SP-CMGs. To challenge the stabilization system, the legs' controller is completely ignorant of the inertial measurement unit (IMU) readings. Moreover, the robot is built around low-end ($2.5k) servo drives. Despite that, our prototype successfully walks blindfolded even through (slightly) rough terrains. To conclude the above, *gyrubot* is a bold little step away from the mainstream developments in legged robotics.

Three main contributions of this work can be summarized as follows:

- The original mechanical design of a 3D bipedal robot based on the idea of robot augmentation with multiple CMGs.
- A synthesis of a naive model-based control. This control is surprisingly viable, even when the legs do not play any part in the stabilization.
- A successful experimental validation of the idea on a real robot.

The rest of the paper is organized as follows: first we provide a brief description of the system (§ II). Then in § III we design two control laws: one for the vertical stabilization and the other one is for the steering of the robot. We provide the experimental results in § IV, and we conclude the presentation in § V.

II. System description

In this section we provide a brief description of the robot shown in Fig. 1.

Legs: Our robot inherits a classical 2 DoF leg design from *RABBIT* [20] and other underactuated robots that have leg actuators located directly at the leg joints. We have minimized the distance between the legs in order to improve its "planarity" and decrease rotational force on the robot (torque in the transverse/frontal planes) during the walking. We have also chosen a passive pinpoint foot design. The leg parts were cut from sheet aluminium and welded together.

A major difficulty in this project was a choice of motors that combine such contradictory qualities as "lightweight", "high torque" and "low cost". For the legs we have chosen four powerful harmonic servo drives RDrive60 with gear ratio 1:100. These motors are capable of providing both high angular velocity (up to $5\,\mathrm{rad/s}$) and a large peak torque ($54\,\mathrm{N\,m}$), which is an essential requirement for ensuring the ability to move quickly while walking.

CMG: The most technically advanced part of our robot is the CMGs. To ease maintenance, we have adopted a modular design: we have four identical easy-to-remove assemblies CMG + actuator. To improve rotational stability at high speeds, the flywheels have been statically and dynamically balanced, and the center of mass of the flywheel-motor system is located at the CMG drive axle. The main flywheel design challenge is to maximize the axial moment of inertia while minimizing the mass, keeping in mind resistance to radial loads and manufacturing constraints. We have performed a FEM analysis of the flywheel deformations to ensure the operational safety at the maximum possible speeds of flywheel rotation and CMG actuation. In addition to that, the flywheel housing is designed in a way to ensure the possibility of full revolution of each CMG without mechanical damage to the structure (with the exception of damage to the wires of the flywheel spinning system) in the event of a failure of the robot software or drive controllers.

For CMG actuation, we have chosen four RDrive50 servomotors by Rozum Robotics for their peak torque of $22\,\mathrm{N\,m}$, mass of about $600\,\mathrm{g}$ and an integrated controller. Four T-Motor MN3110 brushless motors are capable to spin the flywheels up to 12K revolutions-per-minute (RPM) speed. To maintain constant RPM of the flywheels, we have built a custom controller using popular multi-rotor ESC. To have a precise control of the RPM, we have built custom 3 bit magnetic encoders based on Hall sensors. Note that this system is entirely separated from the stabilization system which relies on a constant RPM assumption.

Electronics: During the prototyping and debugging phase, we have used a Speedgoat real-time computer with Simulink environment as the main computational node of the system. The Speedgoat machine communicates with a Nucleo STM32F767 board, the main robot's controller. After completion of the prototyping phase, the Speedgoat computer was removed, and the robot became autonomous, receiving only high-level orders ("start walking", "stop") from an Android tablet.

All servo drives communicate with the Nucleo STM32F767 board via two dedicated CAN channels. To satisfy the bandwidth limitations, we have chosen a $200\,\mathrm{Hz}$ control loop frequency, which is atypically low compared to other bipedal robots [7], [21], [22]. To estimate the inclination of the robot, we have chosen an IMU GyroLab GL203 that communicates with the main board via a serial interface. For the autonomous mode, we use 2 Tattu LiPo $6S$ $10\,\mathrm{A\,h}$ batteries with a total mass of 2.8 kg.

III. Modelling and control

Three independent control loops are used in the robot (refer to Fig. 2 for an illustration):

1) legs' servo drives operate in the open-loop manner with respect to the environment; a pre-computed (model-based) reference signal is repeatedly applied to the servo drives as a feedforward input;
2) flywheels are spun by an autonomous constant-speed controller;
3) finally, four control moment gyroscopes are controlled by taking into account only readings from the IMU and four corresponding encoders, completely ignoring the actual configuration of the legs.

So, all the stabilization of *gyrubot* relies on the CMGs only, the legs do not play any part in it at all.

A. Leg control

While the legs do not take any part in the stabilization of the robot, it can be a good idea to minimize the disturbances

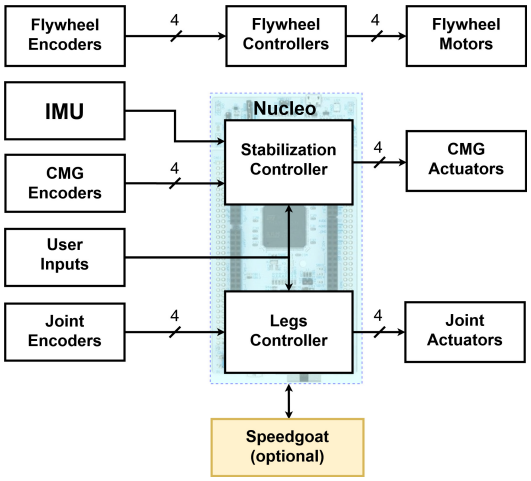


Fig. 2: Three independent control loops are used in the robot. From top to bottom: **(1)** the flywheels spin at a constant speed, **(2)** the CMG actuators react as a function of CMG encoders and IMU readings, **(3)** legs move in constant pre-computed patterns, completely ignoring the environment and the IMU readings.

they bring in. The idea is to pre-compute periodic model-based trajectories for the leg joints that will be repeatedly applied to the legs as a feedforward input. To this end, we build a model of a planar robot using the FROST software [23], that allows us to formulate and solve a non-linear constrained optimization problem. We have put torque and velocity limitations, as well as we have asked for an asymptotically stable trajectory for the robot's torso. The result of the optimization is a set of 4 fifth-order Bézier curves that represent one step of the robot, defining the trajectories for all 4 leg joints.

Note that it is a deliberate choice to ignore the presence of IMU and to control the legs in an open-loop manner w.r.t. the environment. Of course, it would make sense to apply more robust locomotion control methods like the HZD-based control [9], however we have decided to challenge the CMG stabilization system and to implement a very basic control law for the legs. In practice, we use the velocity tracking mode for all leg joints with a PID control loop.

B. Keep it upright

To stabilize itself, the robot uses two scissored pairs of control moment gyroscopes. The idea behind a single-gimbal scissored pair is to spin the flywheels in opposite directions; then, under symmetric control of the gimbals, there is no parasitic torque on the robot's body in the transverse plane. Our scissored pairs are orthogonal and thus the problem of vertical stabilization of the robot can be considered for each axis separately. We therefore consider our robot as two independent rigid inverted pendulums moving in the sagittal and frontal planes, respectively (Fig. 3).

To formulate the control law, first we derive the equations of motion for the system. For each pendulum we choose the state vectors

$$q_1 := [\theta, \alpha_1, \alpha_1', \gamma_1, \gamma_1']^\top \qquad q_2 := [\varphi, \alpha_2, \alpha_2', \gamma_2, \gamma_2']^\top.$$

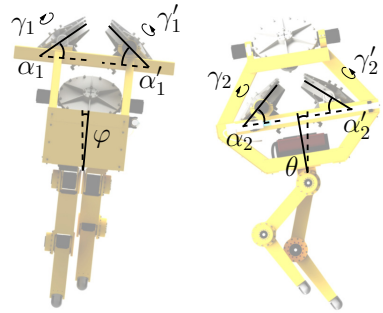


Fig. 3: We consider our robot as two independent inverted pendulums: one moving in the frontal plane (left) and the other moving in the sagittal plane (right). By θ and φ we denote the pitch and roll angles, respectively. Two scissored pair control moment gyroscopes allow to control these angles without interfering with each other. **N.B.** The upper scissored pair (α_1, α_1') controls the pitch θ, whereas the lower pair (α_2, α_2') controls the roll φ.

Throughout this section we use the subscripts $_1$ and $_2$ to denote the sagittal and the frontal pendulums, respectively. Here θ is the pitch, φ is the roll, $\alpha_1, \alpha_1', \alpha_2, \alpha_2'$ denote the CMG gimbal angles and the $\gamma_1, \gamma_1', \gamma_2, \gamma_2'$ denote the angle of the flywheels w.r.t the CMG housing, refer to Fig. 3 for an illustration.

Then we derive the equations of motion for the corresponding Lagrangians $\mathscr{L}_1$ and $\mathscr{L}_2$ (refer to [24], [25] for the full derivation): $\frac{d}{dt}\left(\frac{\partial \mathscr{L}_i}{\partial \dot{q}_i}\right) - \frac{\partial \mathscr{L}_i}{\partial q_i} = \mathbf{S}\tau_i, \quad i = 1,2$, where $\mathbf{S} := \mathrm{diag}(0,1,1,1,1)$ is the selection matrix that determines on which generalized coordinates the actuation torques τ_i act.

Recall that the main idea of a scissored pair control moment gyroscope is to have a symmetric (as symmetric as possible) control of the gimbals in order to avoid parasitic torques in the transverse plane. So, after deriving the equations of motion, we impose the constraints $\alpha_1 = \alpha_1'$ and $\alpha_2 = \alpha_2'$, and remove α_1' and α_2' from the state vectors. In addition to that, we assume that the flywheels spin with a constant speed, so we impose non-holonomic constraints in a form of $\gamma_1(t) = \gamma_1'(t) = \gamma_2(t) = \gamma_2'(t) = \Gamma \cdot t$ and also remove all flywheel angles from the state vectors.

In other words, we redefine the state vectors as: $q_1 := [\theta, \alpha_1]^\top$ and $q_2 := [\varphi, \alpha_2]^\top$, what allows us to represent the equations of motion in matrix form

$$\mathbf{M_i}\,\ddot{q}_i + (\mathbf{C_i} + \mathbf{C_i^{nc}})\,\dot{q}_i + G_i = [0, \tau_i]^\top, \quad i = 1,2, \qquad (1)$$

where $\mathbf{M_i}$ is the mass matrix, $\mathbf{C_i}$ is the Coriolis matrix, G_i is the vector of gravity terms, and $\mathbf{C_1^{nc}}$ and $\mathbf{C_2^{nc}}$ are the matrices of non-conservative forces in the system that in our case are formed by the non-holonomic constraints.

In our hardware the gimbals are controlled by the means of servo drives that ensure velocity tracking. In this case, the velocities $\dot{\alpha}_1$ and $\dot{\alpha}_2$ can be considered as the input signals. We also need to account for the perturbations in the equilibrium position due to the configurations of the legs, external disturbances etc. As proposed in [24]–[26], we add an integral action. To this end, we introduce two auxiliary

variables e_1 and e_2, defined as $\dot{e}_i := \alpha_i,\ i = 1,2$. Then it is convenient to define the new state vectors as follows:

$$x_1 := [\theta, \alpha_1, \dot{\theta}, e_1]^\top \qquad x_2 := [\varphi, \alpha_2, \dot{\varphi}, e_2]^\top.$$

Now let us define two linearization matrices $\mathbf{A_i^{lin}}$ as follows:

$$\mathbf{A_i^{lin}} := \begin{bmatrix} \mathbf{0_{2\times 2}} & \mathbf{I_{2\times 2}} \\ -\mathbf{M_i^{-1}}\frac{\partial G_i}{\partial q_i}(0,0) & -\mathbf{M_i^{-1}}\mathbf{C_i^{nc}}(0,0) \end{bmatrix} \quad i = 1,2$$

Let us denote by $\mathbf{A_i^{lin}}[1\text{–}3, 1\text{–}3]$ the upper left 3×3 submatrix and by $\mathbf{A_i^{lin}}[1\text{–}3, 4]$ first three components of the rightmost column. Then we can write the final linearized equations of motion as follows:

$$\dot{x}_i = \mathbf{A_i} x_i + B_i u_i, \quad i = 1,2, \tag{2}$$

$$\mathbf{A_i} := \left[\begin{array}{c|c} \mathbf{A_i^{lin}}[1\text{–}3, 1\text{–}3] & 0_{3\times 1} \\ \hline 0 \quad 1 \quad 0 & 0 \end{array}\right] \text{ and } B_i := \begin{bmatrix} \mathbf{A_i^{lin}}[1\text{–}3, 4] \\ 0 \end{bmatrix}.$$

To stabilize the pendulums, two linear-quadratic regulators are designed:

$$u_i := -\mathbf{K_i} x_i, \quad i = 1,2, \tag{3}$$

where $\mathbf{K_i}$ is the gain vector minimizing the cost function $\int_0^\infty (x_i^\top(t)\mathbf{Q_i}x_i(t) + R_i u_i^2(t))\,dt$, where the matrices $\mathbf{Q_i} > 0$ and the scalars $R_i > 0$ are the design parameters.

C. Steering

As we will see shortly in the experimental section § IV-A, the control law (3) allows for a quite robust stabilization of the biped w.r.t the vertical. The yaw (transverse plane) angle, however, is not stabilized and the robot gradually "drifts" off the original direction. In this section we present our ongoing work with early results that allow us to alleviate this issue.

As before, we disregard all movements of the legs. The idea is to leave the control law (3) as is for the frontal plane control u_2, and to redefine the control for the sagittal plane pendulum, by incorporating the yaw control. Here we present an *asymmetric* SP-CMG control, therefore, our stabilization system will have three control signals in total.

To synthezise the control law, the general procedure is the same we have presented in § III-B. There are, however, few adjustments. First of all, let us denote by ψ the angle of the robot in the transverse plane. Next we will perform the following change of variables: $\beta := (\alpha_1 + \alpha_1')/2$, $\beta' := (\alpha_1 - \alpha_1')/2$. We define the initial state vector of the system as $q := [\psi,\ \theta,\ \beta,\ \beta', \gamma_1, \gamma_1']^\top$, compute the corresponding Lagrangian $\mathscr{L}$ (refer to [25] for the derivation). Then, we can calculate the Lagrange equations $\frac{d}{dt}\left(\frac{\partial \mathscr{L}}{\partial \dot{q}}\right) - \frac{\partial \mathscr{L}}{\partial q} = \mathbf{S}\tau$, where $S := \mathrm{diag}(0,0,1,1,1,1)$ is the corresponding selection matrix. As before, we impose the non-holonomic constraint $\gamma_1(t) = \gamma_1'(t) = \Gamma \cdot t$ and remove γ_1 and γ_1' from the state vector. It allows us to write the equations of motion in a matrix form:

$$\mathbf{M}\ddot{q} + (\mathbf{C} + \mathbf{C^{nc}})\dot{q} + G = [0,0,\tau,\tau']^\top, \tag{4}$$

where $q := [\psi,\ \theta,\ \beta,\ \beta']^\top$, $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the Coriolis matrix, $\mathbf{C^{nc}}$ is the matrix of non-conservative forces, and G is the vector of gravity terms.

One of the control goals of the Eq. 3 is to keep the CMG gimbals parallel to the ground; it is, however a singular point for the transverse plane control. We therefore shift the goal to some angle δ; in practice we use $\delta = \pi/6$.

Then we define a new linearization matrix $\mathbf{A^{lin}}$ as follows:

$$\mathbf{A^{lin}} := \begin{bmatrix} \mathbf{0_{4\times 4}} & \mathbf{I_{4\times 4}} \\ -\mathbf{M^{-1}}\frac{\partial G}{\partial q}(0,0,\delta,0) & -\mathbf{M^{-1}}\mathbf{C^{nc}}(0,0,\delta,0) \end{bmatrix}$$

To build the control law, as before, we introduce an integral action $\dot{e} := \beta$ and define the state vector as follows:

$$x := [\psi, \theta, \beta, \beta', \dot{\psi}, \dot{\theta}, e]^\top.$$

The linearized dynamics obeys the following law:

$$\dot{x} = \mathbf{A}x + \mathbf{B}\begin{bmatrix} 1/2 & 1/2 \\ 1/2 & -1/2 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix},$$

where the matrices $\mathbf{A}$ and $\mathbf{B}$ are assembled from the corresponding submatrices of $\mathbf{A^{lin}}$, expanded with an integral action, like it is done in § III-B:

$$\mathbf{A} := \left[\begin{array}{c|c} \mathbf{A^{lin}}[1\text{–}6, 1\text{–}6] & \mathbf{0_{6\times 1}} \\ \hline 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 & 0 \end{array}\right],\ \mathbf{B} := \begin{bmatrix} \mathbf{A^{lin}}[1\text{–}6, 7\text{–}8] \\ 0 \quad 0 \end{bmatrix}.$$

Note that the control signals v_1 and v_2 correspond to the real velocities of the servo drives, and a change of basis is needed before they are fed up to the system.

Let us denote by $\bar{\psi}$ the target for the yaw angle that comes from the remote control panel, then the final control law can be expressed as follows:

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} := \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \times (-\mathbf{K}) \times \left(x - [\bar{\psi}, 0, \delta, 0, 0, 0, 0]^\top\right), \tag{5}$$

where $\mathbf{K}$ is a 2×7 gain matrix computed w.r.t the corresponding quadratic cost function. Once again, note the change of basis matrix that connects the linear system with the real *gyrubot*'s input signals.

IV. Experimental results

In this section we present two main experiments: the first one implementing the vertical stabilization control law (3) (§ IV-A) and the other one implementing the control law (5) that allows for steering (§ IV-B). We conclude the section with with our experience on stand-alone (as opposed to external power) walking (§ IV-C).

A. Uneven terrain crossing

Here we present an experiment with *gyrubot* blindly walking through a unknown terrain, refer to Fig. 4 for an action sequence photo. For that purpose, we have built an assembly of randomly laid plywood sheets.

In the beginning of the experiment the robot balances while standing still on both legs. Then *gyrubot* starts marching on the spot (for about 7 seconds) until a signal from the remote control panel is received. Then the robot walks forward for about 15 seconds, first on even ground, then right through the obstacle course that the robot cannot see. After that, having received a command from the remote control panel, the robot marches on the spot (for about 5 seconds) and terminates by standing still. All the phases

TABLE I: Currents during the experiment

	Peak total RMS phase currrent, A		Average total input current, A	
Task	Legs	CMGs	Legs	CMGs
Balancing	2	2	0.01	0.01
Even floor	14	6	1.4	0.7
Obstacle crossing	16	8	1.5	0.8

of the experiment are visible through the plots of the leg joint signals, refer to Fig. 5-a and Fig. 5-b. These plots show clearly that the leg movements are (locally) periodic: the legs move in constant patterns that do not adapt to the environment.

The control goal of the stabilization controller is to keep the robot vertical. Fig. 5-c and Fig. 5-d show the inclination angles as well as the control signals for the control law (3). The plots are color coded: the blue plot of the control signal u_1 corresponds to the blue plot of the pitch angle θ, and the organge plot u_2 corresponds to the orange control goal φ. Note the correlation of these plots with periodic leg movements; it is possible to see all the experiment phases from the inclination signals, including the traversal of the uneven ground.

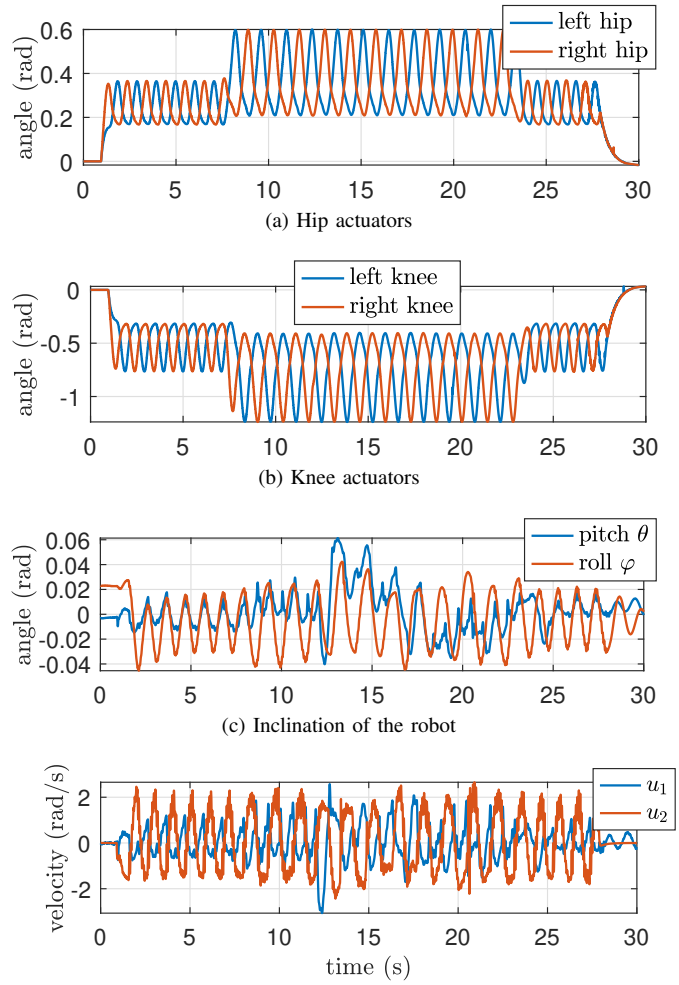
During the experiment the active currents of the *gyrubot*'s actuators varied a lot. The average power consumption of *gyrubot* at a 1 km/h walking gait is about 500 Wt. We performed few identical experiments and measured the peak RMS phase currents as well as the average consumption at different stages of the experiment (balancing while standing still, walking on even terrain, rough terrain crossing). Table I summarizes the measurements. Note that the power consumption of the flywheel motors is almost constant even when the CMGs are heavily actuated.

B. Steering

The second experiment we demonstrate in this section shows the capability of the robot to hold its heading while walking, as well as the tracking of the input reference signal coming from the remote control panel.

In the beginning of the experiment (refer to Fig. 6 and Fig. 7), the robot starts in the vertical stabilization mode

Fig. 4: The robot is able to walk "blindfolded" over small obstacles; no any kind of vision/remote sensing is used. Refer to Fig. 5 for a plot of the signals for this experiment.



(a) Hip actuators

(b) Knee actuators

(c) Inclination of the robot

(d) Control signals

Fig. 5: Plot of the main signals from the experiment shown in Fig. 4. **First and second rows:** hip and knee angles for both legs. Note that these signals present two constant patterns that do not adapt to the environment: marching on the spot and a walking gait. **Third row:** two inclination angles, the control goal is to drive them to zero. **Bottom row:** control signals for both SP-CMGs.

that was described in the previous section. When the steering mode is activated by the operator, the current yaw ψ reading from the IMU is set as a reference for the steering controller. Thus, robort walks forward while holding the direction in the transverse plane.

After several seconds of the robot walking straight, the operator starts to alter the robot's target yaw $\bar{\psi}$ and the effect of the assymmetric control becomes clearly visible! As shown in Fig. 7-c, control inputs v_1 (the blue plot) and v_2 (the orange plot) almost coincide when the reference angle of the robot does not change. However, when the operator drives the robot to a new target, the steering controller forces actuators to move individually, without, of course, forgetting to keep the robot upright, compensating for the gait.

Note that the reference signal $\bar{\psi}$ seen in Fig. 7-a is piecewise constant, this is a direct consequence of the fact

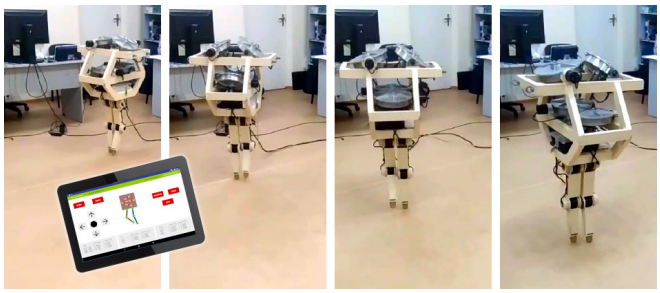


Fig. 6: The steering experiment: with asymettric control of the scissored pairs it is possible to track the yaw angle, allowing for turns. The reference signal is set from a remote control panel. Refer to Fig. 7 for a plot of the signals for this experiment.

that the operator uses a GUI on the remote Android tablet to set the reference.

All phases of the locomotion are presented in Fig. 6 and corresponding plots of the IMU measurements for the entire experiment are presented in Fig. 7-a and Fig. 7-b. We did not plot the leg signals, because they are exactly the same we have already presented in Fig. 5-a and Fig. 5-b. From the yaw angle plot (Fig. 7-c) we can notice oscillations in the transverse plane. We think that the reason for that are the roll angle variations due to the imperfect orientation of the SP-CMG w.r.t the contact point that causes undesirable dynamics of the yaw angle.

C. Getting rid of the leash

During both above experiments the robot had a wire connection and was powered with an external 48 V supply. The principal reason for this wire connection was to record a continuous data stream. Despite that, due to its relatively low power consumption, *gyrubot* is meant to operate autonomously, including outdoors (refer to Fig. 8), receiving high-level control signals from the Android tablet connected via a portable wireless access point. For this purpose two 6S 10 Ah LiPo batteries were added to the system as the onboard power supplies. In these settings, the robot has an operation time a little under 1 h.

Despite the fact that the batteries represent 10% of the total weight of the robot, neither control gains nor leg patterns were recomputed. The robustness of both controllers, synthesized in §-III-B and §-III-C was quite sufficient to execute all the tasks in the real-world operation environment: *gyrubot* successfully balances, walks at a regular sidewalk, tracks the trajectories prescribed by the operator, and even climbs up ramps for disabled persons.

V. Conclusion

The paper introduces *gyrubot*, an underactuated robot augmented with two SP-CMGs capable of blind walking across rough terrain. It was a deliberate choice to challenge the stabilization system with the very basic leg design: it is virtually impossible for a biped robot with 2 DoF legs to walk without external support. To make things even more challenging, we chose to ignore completely IMU readings

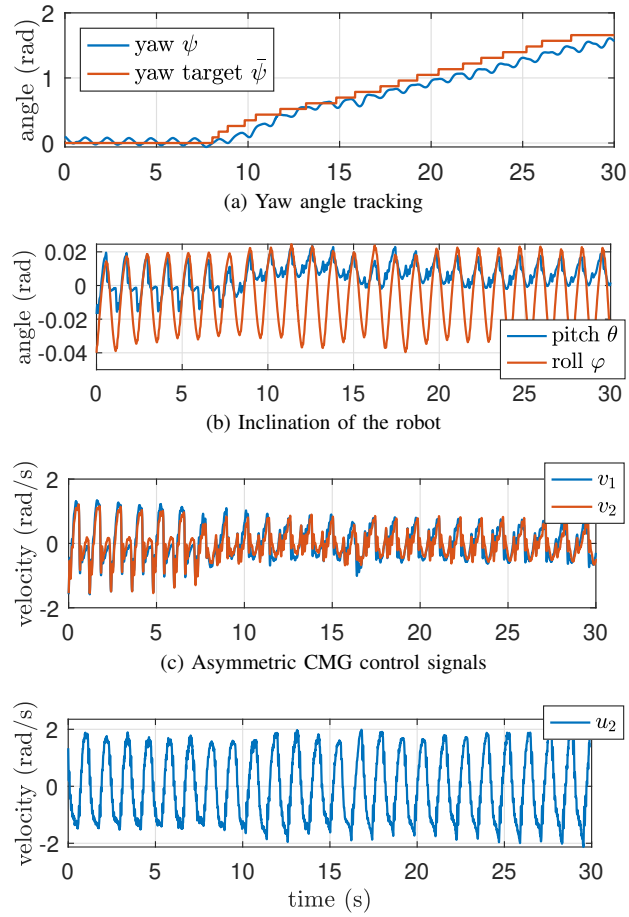


Fig. 7: The steering experiment: plot of the main signals from the experiment shown in Fig. 6. **(a)** the user-provided yaw target $\bar{\psi}$ along with the actual IMU reading of the yaw angle ψ; **(b)** inclination angles of the robot; **(c)** the control law (5); **(d)** the control law (3).

Fig. 8: *gyrubot* is meant to be autonomous, and is capable of walking outdoors. Even loaded with heavy batteries, no new control gains are needed!

for the legs control, they move in an open-loop manner w.r.t the environment. Despite all that, the robot walks in a quite robust manner. Obviously, these exaggerated settings are made for the benchmarking purposes. Auxiliary stabilization systems we propose to build are meant to be *auxiliary*: in real applications legs must do most of the work on the stabilization, and smaller CMGs would give an additional (and not the only!) point of support, providing necessary help in difficult situations.

REFERENCES

[1] Boston dynamics, "The world's most dynamic humanoid robot," 2020. [Online]. Available: https://www.bostondynamics.com/atlas

[2] A. M. Abate, "Mechanical design for robot locomotion," Ph.D. dissertation, 2018.

[3] Agility Robotics, "Meet digit: The newest robot from agility robotics," 2020. [Online]. Available: https://www.agilityrobotics.com/meet-digit

[4] Boston dynamics, "What's new, atlas?" 2017. [Online]. Available: https://www.youtube.com/watch?v=fRj34o4hN4I

[5] S. Chen, J. Rogers, B. Zhang, and K. Sreenath, "Feedback control for autonomous riding of hovershoes by a cassie bipedal robot," 2019.

[6] Boston Dynamics, "Handle, mobile box handling robots for logistics." 2020. [Online]. Available: https://www.bostondynamics.com/handle

[7] V. Klemm, A. Morra, C. Salzmann, F. Tschopp, K. Bodie, L. Gulich, N. Küng, D. Mannhart, C. Pfister, M. Vierneisel, *et al.*, "Ascento: A two-wheeled jumping robot," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7515–7521.

[8] T. L. Brown and J. P. Schmiedeler, "Reaction wheel actuation for improving planar biped walking efficiency," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1290–1297, 2016.

[9] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 42–56, 2003.

[10] I. Ryadchikov, S. Sechenev, E. Nikulchev, M. Drobotenko, A. Svidlov, P. Volkodav, and R. Vishnykov, "Control and stability evaluation of the bipedal walking robot anywalker," *International Review of Automatic Control*, vol. 11, no. 4, pp. 160–165, 2018.

[11] S. Aranovskiy, I. Ryadchikov, E. Nikulchev, J. Wang, and D. Sokolov, "Bias propagation and estimation in homogeneous differentiators for a class of mechanical systems," *IEEE Access*, vol. 8, pp. 19 450–19 459, 2020.

[12] T. C. Wong and Y. S. Hung, "Stabilization of biped dynamic walking using gyroscopic couple," in *Proceedings IEEE International Joint Symposia on Intelligence and Systems*. IEEE, 1996, pp. 102–108.

[13] J. A. Paradiso, "Global steering of single gimballed control moment gyroscopes using a directed search," *Journal of Guidance, Control, and Dynamics*, vol. 15, no. 5, pp. 1236–1244, 1992.

[14] B. Thornton, T. Ura, Y. Nose, and S. Turnock, "Zero-g class underwater robots: Unrestricted attitude control using control moment gyros," *IEEE Journal of Oceanic Engineering*, vol. 32, no. 3, pp. 565–583, 2007.

[15] B. Thornton, T. Ura, and Y. Nose, "Combined energy storage and three-axis attitude control of a gyroscopically actuated auv," in *OCEANS 2008*. IEEE, 2008, pp. 1–9.

[16] J. He and M. Zhao, "Control system design of self-balanced bicycles by control moment gyroscope," in *Proceedings of the 2015 Chinese Intelligent Automation Conference*, Z. Deng and H. Li, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 205–214.

[17] H. Yetkin, S. Kalouche, M. Vernier, G. Colvin, K. Redmill, and U. Ozguner, "Gyroscopic stabilization of an unmanned bicycle," in *2014 American Control Conference*, 2014, pp. 4549–4554.

[18] H. Oya and Y. Fujimoto, "Preliminary experiments for postural control using wearable-cmg," in *IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2017, pp. 7602–7607.

[19] S. Tamayo-León, S. Pulido-Guerrero, and H. Coral-Enriquez, "Self-stabilization of a riderless bicycle with a control moment gyroscope via model-based active disturbance rejection control," in *2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC)*. IEEE, 2017, pp. 1–6.

[20] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. Canudas-De-Wit, and J. W. Grizzle, "Rabbit: a testbed for advanced control theory," *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–79, 2003.

[21] J. W. Grizzle, J. Hurst, B. Morris, H.-W. Park, and K. Sreenath, "Mabel, a new robotic bipedal walker and runner," in *2009 American Control Conference*. IEEE, 2009, pp. 2030–2036.

[22] G. Nelson, A. Saunders, N. Neville, B. Swilling, J. Bondaryk, D. Billings, C. Lee, R. Playter, and M. Raibert, "Petman: A humanoid robot for testing chemical protective clothing," *Journal of the Robotics Society of Japan*, vol. 30, no. 4, pp. 372–377, 2012.

[23] A. Hereid and A. D. Ames, "Frost*: Fast robot optimization and simulation toolkit," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 719–726.

[24] D. Sokolov, S. Aranovskiy, A. Gusev, and I. Ryadchikov, "Experimental comparison of velocity estimators for a control moment gyroscope inverted pendulum," in *2020 IEEE 16th International Workshop on Advanced Motion Control (AMC)*, 2020.

[25] S. Aranovskiy, I. Ryadchikov, E. Nikulchev, J. Wang, and D. Sokolov, "Experimental comparison of velocity observers: A scissored pair control moment gyroscope case study," *IEEE Access*, vol. 8, pp. 21 694–21 702, 2020.

[26] I. Ryadchikov, S. Sechenev, N. Mikhalkov, A. Biryuk, A. Svidlov, A. Gusev, D. Sokolov, and E. Nikulchev, "Feedback control with equilibrium revision for cmg-actuated inverted pendulum," in *Proceedings of 14th International Conference on Electromechanics and Robotics "Zavalishin's Readings"*, A. Ronzhin and V. Shishlakov, Eds. Singapore: Springer Singapore, 2020, pp. 431–440.