1. Introduction

In this paper a control algorithm for a hopping robot having one articulated leg is presented and a description of the mechanical design of a prototype, constructed at the Department of Mechanical Engineering of the VUB, is given. The control algorithm enables the robot, whose name is OLIE (One Leg Is Enough), to control simultaneously its forward velocity during flight, its hopping height, the placement of its foot on desired footholds, the orientation of its body and its angular momentum.

The model of the hopping robot under study and its control algorithm are given in section 2. Simulation results are presented and remarks are made in section 3. In section 4 a description is given of the design of the experimental prototype of the robot. Finally, in section 5, conclusions are drawn and comments on further research are made.

2. A hopping robot with one articulated leg

2.1. The model
The control of a running machine is more critical than the control of a walking machine. To be able to study all the features of a running machine, such as its underactuated and nonholonomic nature, without unnecessarily increasing the complexity of its design, a robot having one articulated leg is considered here. For the same reason its motion is restricted to the sagittal plane.

Figure 1 depicts the robot at the moment of take-off, denoted by the superscript $\text{to}$, and at the moment of touch-down, denoted by the superscript $\text{td}$. The robot consists of three segments: a lower leg (segment 1), an upper leg (segment 2) and a body (segment 3). The different links are connected to each other by actuated rotational joints.

The length of the $i$-th link is $l_i$, its mass is $m_i$ and the moment of inertia around its center of mass $G_i$ is $I_i$. The angle between the horizontal and the $i$-th segment is $\theta_i$. Point $F$ will further be referred to as the foot of the robot, point $K$ as its knee and point $H$ as its hip. The center of mass $G_3$ of the body coincides with the hip.

The ground is modelled, in the horizontal and in the vertical direction, as a parallel spring-damper combination. This allows to simulate the flight phase, during which there is no contact with the ground, the impact phase, when the foot hits the ground, and the stance phase, during which the robot stands on the ground, all without mathematically changing the number of degrees of freedom.
(DOF) of the system. However, by tuning the spring constants and the damping coefficients it is possible to keep the displacement of the foot during stance arbitrarily small, thus virtually reducing the number of DOF of the robot.

Considering the previous remark the robot has 5 DOF during flight and 3 DOF during stance (the assumption has been made that the foot of the robot does not slip). The generalized coordinates used to describe the motion of the robot are the coordinates $x_F$ and $y_F$ of the foot, measured in the reference frame OXYZ, and the angles $\theta_1$, $\theta_2$, and $\theta_3$.

2.2. The control algorithm

To allow the motion of a legged machine to be flexible in terms of locomotion criteria or goals to be reached, as opposed to steady-state behaviour, it is necessary to take as control variables those parameters that are linked to the desired behaviour as closely as possible. When moving on irregular terrain for example, it makes more sense to express the locomotion pattern in terms of the coordinates of the feet than in terms of the internal angles between the different links of the machine. The control variables used should describe the motion from an external point of view, instead from an internal one, which leads to the concept of objective locomotion parameters. Expressing locomotion in function of objective parameters is a strategy commonly used in the domain of animation [2][3]. The control algorithm presented here is based on the objective parameters approach, making locomotion on irregular terrain possible and thus broadening locomotion limits such as encountered in [4]. The algorithm allows a hopping robot to simultaneously control its forward speed during flight, its hopping height (defined as the difference between the maximum height of $G$ during flight and its height at take-off), the placement of its foot on desired footholds, the orientation of its body and its angular momentum. The desired values of all these objective locomotion parameters can be altered from one hop to another.

Since there are only two actuators the robot is an underactuated system, subject to a number of holonomic and nonholonomic constraints. When compared to classical robots a different kind of control is needed. The general idea of the control algorithm presented in this paper, is to steer the dynamics of the robot in a kinematic way, resulting in dynamically stable behaviour. In function of the
settings of the objective parameters that have to be controlled, a unique set of take-off and landing conditions can be calculated that comply with the linear and angular momentum constraints. Once these boundary conditions are known, trajectories for the different joint angles of the robot can be generated that, when tracked by a controller, guarantee that the locomotion goals are met. Although it is not a necessary condition for this strategy to work, decoupling the motion of the global center of mass $G$ of the robot from the rotation of the robot’s body, i.e. by placing the center of mass $G$ of the body in the hip, greatly reduces complexity and results in a control algorithm consisting of two main parts, decoupled as well. By steering the motion of the leg the first part controls the robot’s forward velocity during flight, its hopping height and the placement of its foot at touch-down. The second part controls the orientation of the body and the robot’s angular momentum.

2.2.1. Part 1: Control of the forward velocity during flight, the hopping height and the placement of the foot

The core of the first part of the control algorithm focuses on the flight phase and is based on the kinematics of the robot on one hand:

$$\begin{align*}
\mathbf{r}_G &= \mathbf{r}_G(x_F, y_F, \theta_1, \theta_2) \\
\mathbf{\dot{r}}_G &= \mathbf{\dot{r}}_G(\mathbf{x}_F, \mathbf{y}_F, \theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) \\
\mathbf{\ddot{r}}_G &= \mathbf{\ddot{r}}_G(\mathbf{x}_F, \mathbf{y}_F, \theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_1, \dot{\theta}_2)
\end{align*}$$

(1)

and the dynamics of the robot, being the equations dictating the parabolic trajectory of $G$, on the other hand:

$$\begin{align*}
\mathbf{\dddot{r}}_G &= \mathbf{\dddot{r}}_G(g, x_F^{t^0}, y_F^{t^0}, \mathbf{\dot{x}}_G^{t^0}, \mathbf{\dot{y}}_G^{t^0}, \dot{t}^{t^0}, t) \\
\mathbf{\dddot{r}}_G &= \mathbf{\dddot{r}}_G(g, \mathbf{\dot{x}}_G^{t^0}, \mathbf{\dot{y}}_G^{t^0}, \dot{t}^{t^0}, t) \\
\mathbf{\dddot{r}}_G &= \mathbf{\dddot{r}}_G(g)
\end{align*}$$

(2)

with $\mathbf{r}_G$ being the position vector of $G$, $g$ being the gravitational acceleration and $t$ being time.
Assuming the foot is stationary during stance and choosing its position as the origin of the reference frame yields:

\[ x_{F}^{i} = y_{F}^{i} = \theta_{1}^{i} = \theta_{2}^{i} = \dot{\theta}_{1}^{i} = \dot{\theta}_{2}^{i} = 0 \]  \( \text{(3)} \)

Expressing that at touch-down the orientation of the leg should be fixed, gives:

\[ \theta_{1}^{cd} = \theta_{2}^{cd} = \theta_{1}^{d} = \theta_{2}^{d} = 0 \]  \( \text{(4)} \)

Choosing the orientation of the lower leg at take-off, demanding that the knee-angle at touch-down equals the knee-angle at take-off and demanding that the CG-print \(^{5}\) is symmetrical with respect to the vertical when comparing the configuration at touch-down to the configuration at take-off, results in three constraints of the form:

\[ \phi_{i}(\theta_{1}^{io}, \theta_{2}^{io}, \theta_{1}^{id}, \theta_{2}^{id}) = 0 \quad i = 1, \ldots, 3 \]  \( \text{(5)} \)

Setting the subset of objective locomotion parameters, consisting of the forward velocity \( \dot{\theta}_{1} \) of the center of mass during flight, the jumping height \( \Delta h \), the horizontal position \( x_{F}^{td} \) of the foot at touch-down and the vertical position \( y_{F}^{td} \) of the foot at touch-down, equal to its desired value (denoted by the superscript \(^{*}\)), gives:

\[
\begin{align*}
\dot{\theta}_{1}^{io} &= \dot{\theta}_{1}^{o} \\
\dot{\theta}_{2}^{io} &= \dot{\theta}_{2}^{o} \\
\dot{\theta}_{1}^{id} &= \dot{\theta}_{1}^{d} \\
\dot{\theta}_{2}^{id} &= \dot{\theta}_{2}^{d} \\
x_{F}^{td} &= x_{F}^{*} \\
y_{F}^{td} &= y_{F}^{*}
\end{align*}
\]  \( \text{(6)} \)

By identifying equations (1) and (2) on the positional level at touch-down, by identifying the same equations on the velocity and acceleration level at take-off and by imposing the boundary conditions (3) to (6), the angles \( \theta_{1} \) and \( \theta_{2} \) needed at take-off and at touch-down, as well as their corresponding angular velocities and angular accelerations, are calculated. Next, nominal trajectories for \( \theta_{1} \) and \( \theta_{2} \)
are generated, for both the flight phase (denoted by the superscript $^{\text{fl}}$) and the stance phase (denoted by the superscript $^{\text{st}}$), satisfying the conditions at take-off and touch-down:

$$
\begin{align*}
\theta_1^{\text{fl}} &= \theta_1^{\text{nom,fl}}(t) \\
\theta_2^{\text{fl}} &= \theta_2^{\text{nom,fl}}(t) \\
\theta_1^{\text{st}} &= \theta_1^{\text{nom,st}}(t) \\
\theta_2^{\text{st}} &= \theta_2^{\text{nom,st}}(t)
\end{align*}
$$

(7)

The motor actuating the knee then tracks the trajectory for $\theta_1$, the motor actuating the hip tracks the trajectory for $\theta_2$. When the prescribed trajectories are tracked with a sufficiently small error, the robot is theoretically able to perform any desired hopping pattern, with every hop satisfying its forward velocity as well as its desired hopping height, and with the foot landing on the desired footholds.

### 2.2.2. Part 2: Control of the orientation of the body and the robot’s angular momentum

Since the angle $\theta_3$ between the body and the horizontal does not appear in the expressions giving the trajectory of $G$, the first part of the control algorithm is not able to control the body’s orientation. Starting from an initial orientation the body will start rotating and will keep on doing so. The angular momentum shows the same drift. To prevent destabilization of the locomotion of the robot due to this behaviour, a second part in the control strategy is needed.

Therefore the nonholonomic constraints expressing that the angular momentum $\mu_G^{\text{fl}}$ around $G$ during flight is equal to a constant $C$ and that the angular momentum $\mu_F^{\text{st}}$ around the foot during stance only depends on gravity, have to be taken into account:

$$
\begin{align*}
\mu_G^{\text{fl}} &= f_G(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) + I_1 \dot{\theta}_3 = C \\
\frac{d}{dt} \mu_F^{\text{st}} &= \frac{d}{dt} \left[ f_F(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) + I_1 \dot{\theta}_3 \right] = g_F(g, \theta_1, \theta_2)
\end{align*}
$$

(8) (9)

with $f_G$, $f_F$ and $g_F$ being specific functions only depending on the state of the leg.
Integrating (9) once and twice over the stance phase and combining the result with (8), expressed at touch-down and at take-off, gives:

\[
\mu_G^{to} = \int_{t_1}^{t_2} f_g(\theta_1^{st}, \theta_2^{st}) \, dt + \left[ f(\theta_1^{st}, \theta_2^{st}, \dot{\theta}_1^{st}, \dot{\theta}_2^{st}) \right]_{t_1}^{t_2} + \mu_G^{td}
\]

\[
\theta_3^{to} = \int_{t_1}^{t_2} \left[ g_f(\theta_1^{st}, \theta_2^{st})(T^{st} - t) - f_f(\theta_1^{st}, \theta_2^{st}, \dot{\theta}_1^{st}, \dot{\theta}_2^{st}) \right] \, dt + \mu_G^{td} T^{st} \right\} I_3 + \theta_3^{td}
\]

with f being a specific function determined by the state of the leg and \( T^{st} \) being the stance time.

To change the values of the angular momentum \( \mu_G^{to} \) and the angle \( \theta_3^{to} \), correction functions for the angles \( \theta_1 \) and \( \theta_2 \) during stance, depending on two new parameters \( C_1 \) and \( C_2 \), are introduced:

\[
\theta_1^{corr,st} (t) = C_1 t^3 \left( T^{st} - t \right)^3
\]

\[
\theta_2^{corr,st} (t) = C_2 t^3 \left( T^{st} - t \right)^3
\]

By adding the correction functions to the nominal trajectories:

\[
\theta_1^{st} = \theta_1^{nom,st} (t) + \theta_1^{corr,st} (t)
\]

\[
\theta_2^{st} = \theta_2^{nom,st} (t) + \theta_2^{corr,st} (t)
\]

and by using (10) and (11), \( C_1 \) and \( C_2 \) can be calculated such that the angular momentum \( \mu_G^{to} \) and the angle \( \theta_3^{to} \) are equal to their desired values:

\[
\mu_G^{to} = \mu_G^{to,*}
\]

\[
\theta_3^{to} = \theta_3^{to,*}
\]

The actuators can now track the new trajectories. Since the correction functions respect the boundary conditions calculated in the first part of the control algorithm:
the orientation of the body of the robot is controlled, without altering the robot’s forward velocity, its
hopping height or the placement of its foot.

3. Simulation results and remarks

To test the algorithm described above a hopping pattern consisting of two consecutive hops has been
simulated. The parameters of the simulated robot are given in table 1. The values of the desired
objectives, being the forward velocity of the center of mass during flight, the hopping height, the step
length, the stepping height, the orientation of the body at take-off and the angular momentum around
G at take-off, are the same for both hops and are given in table 2. The robot hops on horizontal
terrain, which is expressed by the fact that the desired stepping height equals zero.

Some simulation results are shown in figures 2 to 5.

<table>
<thead>
<tr>
<th>Table 1: model parameters</th>
<th>Table 2: desired objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l_1 = 0.3 \ m$</td>
<td>$\mathbf{v}_c = 1.0 \ m/s$</td>
</tr>
<tr>
<td>$l_2 = 0.3 \ m$</td>
<td>$\Delta h^* = 0.15 \ m$</td>
</tr>
<tr>
<td>$m_1 = 1.0 \ kg$</td>
<td>$\Delta x_f^* = 0.5 \ m$</td>
</tr>
<tr>
<td>$m_2 = 1.0 \ kg$</td>
<td>$\Delta y_f^* = 0.0 \ m$</td>
</tr>
<tr>
<td>$I_1 = 0.3 \ kgm^2$</td>
<td>$\theta_3^{to} = 0.0 \ rad$</td>
</tr>
<tr>
<td>$I_2 = 0.3 \ kgm^2$</td>
<td>$\mu_G^{to} = 0.0 \ kgm^2/s$</td>
</tr>
<tr>
<td>$l_3 = 0.6 \ m$</td>
<td></td>
</tr>
<tr>
<td>$m_3 = 8.0 \ kg$</td>
<td></td>
</tr>
<tr>
<td>$I_3 = 0.6 \ kgm^2$</td>
<td></td>
</tr>
</tbody>
</table>
Figure 2 shows the forward velocity \( \mathbf{v}_g \) of the global center of mass versus time. The horizontal parts of the graph give the velocity during the flight phases, equaling the desired value. Figure 3 shows the horizontal position \( x_f \) of the foot versus time, with the horizontal parts in the graph representing the position during the stance phases. The step length is given by the distance between two successive horizontal parts and is equal to the desired value. The vertical position \( y_G \) of the global center of mass of the robot is given in figure 4. The horizontal line at the bottom indicates the height of \( G \) at take-off. The other horizontal line indicates the maximum height, i.e. the height when the top of the parabolic trajectory of \( G \) during flight is reached. The distance between the two lines gives the hopping height, being equal to the desired value. Figure 5 represents the orientation of the body \( \theta_3 \) versus time. Without the second part of the control strategy the drift in the body’s orientation is clearly visible. With the extra control added the body is positioned horizontally at take-off.

Figures 2 to 5 show that the control strategy presented in this paper works very efficiently. Controlled periodic or aperiodic hopping can be achieved, making it possible for the robot to hop over obstacles, along a path with randomly distributed footholds and at a desired speed, hereby controlling the orientation of its body.

A remark can be made however, concerning the physical admissibility of the simulated motion of the robot. It is clear that high accelerations and thus high actuator torques can result from a specific combination of desired objective parameters. This means that due to actuator limits and due to the geometry of the robot not every combination of desired objectives will be realizable in practice.

4. Experimental prototype
4.1. Mechanical layout

OLIE is the electrically actuated prototype of the robot described above, its mechanical layout being given in figure 6. Its total weight is 11.66 kg and its height, when standing in an upright position, is 0.65 m.

Fig. 6: mechanical layout

The upper and lower leg of the robot are built up by aluminium and wooden parts in order to get a lightweight construction. The upper body is an aluminium frame, horizontally placed upon the leg. A rubber cushion located at the bottom of the lower leg is referred to as the foot.

Two carbon steel torsional springs, forming the passive part of the robot’s actuation, are placed at the knee, exerting a torque as a function of the relative angle between upper and lower leg.

Two electromotors actuating hip and knee respectively are located at the ends of the upper body. This configuration assures a high ratio of body inertia to leg inertia, so that during flight body pitching is minimised.

To transfer the torques produced by the motors to the corresponding joint, and to transform the high speed-low torque characteristics of the motors to the low speeds and high torques necessary to
drive the robot, two Synchroflex T5 toothed belt transmission loops were introduced. Each belt loop is formed by two sub-loops resulting in a total gear ratio of 1:20 for both the hip and the knee. Of each link the length \( l \), the mass \( m \) and the moment of inertia \( I_{\text{com}} \) around the center of mass are given in table 3.

<table>
<thead>
<tr>
<th></th>
<th>( l ) ((m))</th>
<th>( m ) ((kg))</th>
<th>( I_{\text{com}} ) ((kgm^2))</th>
</tr>
</thead>
<tbody>
<tr>
<td>body</td>
<td>0.666</td>
<td>8.507</td>
<td>0.7979</td>
</tr>
<tr>
<td>upper leg</td>
<td>0.308</td>
<td>1.373</td>
<td>0.0218</td>
</tr>
<tr>
<td>lower leg</td>
<td>0.342</td>
<td>1.781</td>
<td>0.0138</td>
</tr>
</tbody>
</table>

The robot’s motion is restricted to the 2D surface of a cylinder by a boom mechanism, shown in figure 7. This mechanism, which is connected to the robot at the hip, permits the machine to translate horizontally and vertically, and to rotate around the axis of the boom. The boom is made of a lightweight composite tube, having negligible weight and moment of inertia. Its length is variable between 2 \( m \) and 3 \( m \).

4.2. Actuators
Although the power to weight ratio of electrical actuators is lower when compared to hydraulic actuators, they make use of an energy source which is far more available. In addition, they can be integrated more easily in the machine and they are better steerable [6]. In order to obtain electrical actuation with best power to weight and torque to weight ratio, brushless AC servomotors were chosen.

Preliminary simulations indicated that the minimum values for torque and power supply at hip and knee were 30 Nm and 300 W respectively. Further limiting resulted in not reaching the desired values of the objective locomotion parameters. After introducing a safety factor to overcome discrepancies between the simulations and the physical robot, such as internal friction in the joints and extra power absorbed by the transmission, two SEM HD55G4-44S AC brushless motors were chosen. These motors are steered by IRT BR1306 AC-servo-controllers. Combined with a gear ratio of 1:20 for the belt transmission, the actuators generate a peak stall torque of 58 Nm at both the hip and the knee. Maximum power supply is approximately 1.2 kW, one motor weighing only 1.9 kg.

Each of the torsional springs forming the passive part of the actuation, has a spring constant of 45 Nm/rad and a mass of 0.186 kg.

4.3. Sensors

In order to provide measurements of the absolute position of the robot, two sensors are mounted on the boom pivot. A multi-turn potentiometer is used to determine the horizontal position of the robot’s hip, while a single-turn potentiometer measuring the angle between boom and pivot, is used to determine the vertical position of the hip.

Three more sensors mounted on the robot yield the orientation of the different links: two potentiometers measure the relative angles at the hip and the knee respectively, while another potentiometer measures the relative angle between the upper body and the boom, about the axis of the boom. A mechanical switch located at the foot senses contact with the ground.

Except for the multi-turn unit which uses a wirewound resistance element, all potentiometers used on the robot consist of conductive plastic resistance elements. When compared to other types, the
Conductive plastic units are more shock-resistant and have better electrical noise characteristics. Linearity is better and life of a conductive plastic unit is longer than that of other types.

4.4. Control scheme

The direction, position and speed of the actuator shafts are controlled by DC voltages applied to the servocontrollers. All calculations concerning the control are performed in LABVIEW on a Pentium-S PC. The link between the PC and the servocontroller on one hand and between the potentiometers on the robot and the PC on the other hand is formed by a National Instruments AT-MIO-16E-10 data acquisition board.

A schematic representation of the control system for one actuator is given in figure 8. The angle $\theta_R$ is the actual value of the relative angle in a joint of the robot being measured by a potentiometer and $S$ is a Boolean variable whose value depends on the state of the contact-switch on the foot. Each time the value of $S$ changes, which happens whenever a transition from flight to stance or vice versa occurs, the control algorithm described above calculates the desired trajectory $\theta_R^*$, which has to be tracked during the next flight or stance phase in order to obtain the desired values of the objective parameters. During the simulations this angle was PD-controlled. In the experimental model the BR 1306 servocontroller input expects a DC voltage $V_A$ (-10V ≤ $V_A$ ≤ +10V). This voltage is transformed by the servocontroller to the desired value of the shaft speed $n_A$, which is PI-controlled. This means that in fact only the trajectory for the angular velocity $\dot{\theta}_R^*$ of the link of the robot can be tracked, and that an uncontrolled error between the desired value of the angle $\theta_R^*$ and the actual value $\theta_R$ measured by the potentiometer can occur. To avoid this problem, a correction term $\Delta V_A$ is added to the voltage $V_A$. This correction term is determined by a PI-controller which acts on the signals $\theta_R$ and $\theta_R^*$. In this way, deviations from the desired value of the angle $\theta_R^*$ are being removed by adding small corrections to the trajectory of $\theta_R$. 
5. Conclusion

An algorithm is presented enabling a one-legged hopping robot to control simultaneously its forward velocity during flight, its hopping height, its step length, its stepping height, the orientation of its body and its angular momentum. Simulation results show that the algorithm is efficient and that the desired objectives are attained. A robot using this algorithm should be able to jump over obstacles and to move over terrain where footholds are randomly distributed.

A mechanical prototype of the simulated robot has been built and its design is presented. The active part of the robot’s actuation consists of two electromotors, the passive part of two springs in the knee.

On the theoretical level future research will focus on the robustness of the control algorithm and on the determination of the physically admissible subspace, in terms of actuator and geometrical constraints, of the space formed by the controlled parameters. On the practical level the control strategy will be implemented on the mechanical prototype and the experimental results will be compared to the simulated results.

References


