Proprioceptive Control of a Hybrid Legged-Wheeled Robot
Markus Eich, Felix Grimminger, Frank Kirchner
DFKI Bremen
(German Research Center For Artificial Intelligence)
Robotics Group
28359 Bremen, Germany
{first_name.last_name}@dfki.de

Abstract—In this work we describe an innovative proprioceptive control architecture for our hybrid legged-wheeled robot ASGUARD. This robot is able to cope with a variety of stairs, very rough terrain, and is able to move with the speed of two body-lengths per second on flat ground. An additional proprioceptive inclination feedback is used to make the same controller more robust in terms of stair-climbing capabilities. Contrary to existing approaches, we did not use a pre-defined walking pattern for stair climbing, but an adaptive approach based only on internal sensor information. The data we use in our architecture is based on proprioceptive information, like body inclination and external torques, which are acting on the driving motors. In this work we show how this adaptivity results in a versatile controller for hybrid legged-wheeled robots. For the locomotion control we use an adaptive model of motion pattern generators. In contrast to many other walking pattern based robots, we use the direct proprioceptive feedback in order to modify the internal control loop, thus adapting the compliance of each leg on-line. For different terrains and stairs we use a phase-adaptive pattern which is using directly the proprioceptive data from each leg. We show that our adaptive controller is able to improve the stair-climbing behaviour in terms of energy consumption and energy distribution.

Index Terms—adaptive robot locomotion; legged wheel; pattern generator; stair climbing

I. INTRODUCTION

In [1], [2] we presented for the first time our fast and highly agile quadruped robot ASGUARD. The robot was designed to be used in harsh outdoor environment with a focus on security and outdoor surveillance as well as on disaster mitigation missions. For those applications, a robot has to transport a variety of mission-depending application sensors inside a difficult terrain. Those missions are commonly named “Three D” missions. “Three D” stands for dull, dirty, and dangerous and implies, e.g., tasks where rescue personnel must enter a collapse-endangered building in search for injured people, the acquisition of samples in contaminated areas, or patrolling every day along the same fence of a security-relevant compound. For all those applications, an in-situ system has to deal with obstacles or uneven and difficult outdoor terrain. Additionally, the robot should be able to move fast where the ground is levelled and easier to cross. In order to cope with those two requirements, the quadruped robot ASGUARD was designed. It makes use of multiple rotating legs along one hip shaft which are controlled by four individual motion pattern generators, controlling each leg in time-angle-space. Figure 1 shows the robot in a typical outdoor environment.

The concept of using motion pattern is well known and utilized in the area of ambulating robots. An example for generated motion pattern are Central Pattern Generators (CPGs) as the major mechanisms in animals to control and to produce rhythmic motion. CPGs are characterized by the ability to produce rhythmic motion patterns via oscillation of neuronal activity without the need of sensory feedback [3]. However, sensory feedback is normally integrated into the CPGs. Mostly load and position data of the controlled limb/joint are fed back into the CPG-network which is used to implement a closed-loop control of the rhythmic motion of the system actuators. To modulate the controlled rhythmic patterns, the CPG can change its frequency, phase, and amplitude [4].

For the use in robots it is reasonable to develop an abstract CPG model which inherits only the basic principles of the CPG’s functionality. Many different ways to achieve this have been proposed and tested, e.g. [5]–[8]. In [9] our approaches for controlling walking robots are described in detail. Besides multi-joint walking machines, pattern-based ap-
proaches are widely used in hybrid locomotion. In [10], for instance, the hexapod RHEX is described. The robot uses one rotating compliant leg per actuator and is able to ascend and descend stairs. RHEX uses a fixed pattern for the trajectory of each leg. The locomotion is performed by a tripod gait, where the retraction and protraction phases are alternatingly triggered. For the stair-climbing behaviour of the RHEX robot, six phases were defined, based on a fixed transition model [11], [12]. Another proprioceptive information of the robot is used in [13], where the motor current is used to detect the contact with a flight of stairs. In this case, the tripod gait changes to a metachronal wave gait. RHEX uses fixed gait transitions, but in contrast to our approach, the trajectories of the legs are often based on a fixed, hand-adjusted gait configuration. A similar control approach as in RHEX can be found in [14], but uses a tri-lobe wheel to implement a quadruped locomotion. [15] and [16] use a design of a multi-spoked wheel for their hexapod WHEGS, which comes closer to the design of our quadruped ASGUA RD because WHEGS uses more than one compliant leg per axis and is able to adapt its walking pattern by pure mechanical compliance of the legs. The bio-inspired mechanical design is derived from an analysis of the cockroach gait. WHEGS uses no sensor information to adjust the tripod gait: it uses only the compliant legs design to adapt to different types of terrain. WHEGS uses only one DC motor for locomotion and one servo for active steering.

To our knowledge, all locomotion concepts for hybrid legged-wheeled approaches are based on fixed motion patterns. Inclination and ground contact are usually used to select from a range of predefined walking patterns. Only WHEGS uses an adaptation based on pure mechanical compliance of the legs, but this compliance cannot be changed on-line during locomotion.

Our robust stair-climbing behaviour is based on an adaptive closed-loop approach where the direct torque feedback is used to adapt the overall walking pattern. While running fast on flat ground, e.g. a paved road, the legs of the robot must not be compliant. Therefore we implemented two different controllers for the robot: one for flat ground and one for stair climbing and very rough terrain. Therefore we included an inclination sensor, on which the output of the two different controllers are gradually merged.

The remainder of the paper is arranged as follows: a short summary of the physical platform ASGUA RD is described in Section II. The general control concept and its latest extension is described in Section III. In Section IV we present the experimental results of our approach. In Section V we will discuss those and give some ideas about our future research.

II. ASGUA RD PLATFORM

The long-term goal of our research is to develop a robust outdoor platform which is suitable to be included in disaster mitigation as well as in security and surveillance missions. The platform should be able to transport application sensors to areas that are dangerous for humans to access, e.g. a collapse-endangered building or an industrial compound after a chemical accident. In those cases, before they enter, the rescue personnel might need some information about the air contamination or the whereabouts of people inside an area. The robot should be upgradeable with a variety of application sensors, e.g. cameras, thermal vision, or chemical sensors. To be usable in any search and rescue or security application, the robot has to be operational without changing batteries for at least two hours. All these requirements were specified in cooperation with potential end users, like fire fighters and rescue personnel. This defined the minimum size of ASGUA RD, as well as the energy budget and the minimum payload. To be usable for a variety of missions, the robot has to be able to carry sensors to areas which are normally not accessible to wheeled and tracked robots.

The robot ASGUA RD is a hybrid quadruped outdoor robot using only proprioceptive data which come from the system itself to adapt its walking behaviour. The first prototype of our system is driven by four directly actuated legs with one rotational degree of freedom. In Figure 2 the robot is climbing a stair on our test track. After testing the ground traction with a rigid corpus, we found out that we could increase ground contact by adding an additional rotational degree of freedom along the body axis, serving as an elastic spinal column. By this we could increase the ground traction significantly. The spinal column is restricted in movement between $-40^\circ$ and $40^\circ$.

For the low-level control of the robot, a custom-designed FPGA motor control board (MOTCON6) is used which controls the four motors in a closed-loop manner. The locomotion of the robot is performed by four independent pattern generators which describe the trajectory of each leg within the phase of $[-\frac{1}{5}\pi, \frac{1}{5}\pi]$. With the MOTCON6 board, we are also able to measure...
the power consumption as well as the position of each leg in real time, providing important proprioceptive information about the system. In contrast to other approaches, for each leg we can individually define the trajectory, allowing us to synchronize the legs with each other, or to shift the phase of each leg trajectory.

The compliant legs of the robot are arranged around four hip shafts, with an angular distance of $\frac{2\pi}{5}$. Because of the symmetry of the legs, we have only to consider the phase between $[-\frac{1}{5}\pi, \frac{1}{5}\pi]$ (cf. Figure 3). This configuration makes sure that we have a minimum of four legs on the ground, which ensures a stable configuration of the robot. The outer radius of the legged wheel is 22 cm. The inner radius (i.e. the height of the hip joint shaft if two legs have ground contact) of the legged wheel is 18 cm. In order to decrease the physical shock during locomotion, shock-absorbing leg tips were used.

While driving with high velocities, only the leg tips have direct contact to the ground. In this case, ASGUARD behaves like a wheeled system, with the radius $b$ (cf. Figure 3), reaching velocities of around $2 m/s$ which is equivalent to two body-lengths per second.

III. ADAPTIVE LOCOMOTION CONTROL

A. Control of Hybrid Legged Wheels

In order to control our robot ASGUARD, we are facing two requirements. On one hand, we have to control the velocity, i.e. the rotational speed of each of the legs. On the other hand, we have to control the position of the four legs of the robot. For controlled stair climbing, we have to keep track of the leg positions over time.

From the CPG control methods, used in a variety of walking machines, we developed an efficient approach to control such systems by using trajectories in the time-angle space. In contrast to many pure legged robots, which have generally more than one degree of freedom for each leg, we have only one angle to control over time. Moreover, as described in Section II, we only have to consider the angular space between $[-\frac{1}{5}\pi, \frac{1}{5}\pi]$.

For the directional and speed control of ASGUARD, a high level controller, which receives its input directly via a joystick, sends the parameters for phase, frequency, and direction to the trajectories (cf. Figure 4). From our high-level controller we can modify the pattern parameters by changing the pattern frequency, the direction of the generated pattern as well as a phase offset. By this phase offset we can change the synchronization of each of the legs. In our case, a sawtooth pattern is used which can be controlled individually in phase, frequency, and amplitude for each leg.

The patterns are then generated in parallel on the FPGA board. A position controller, which is on the next lower level in our architecture, is also implemented in the FPGA. Its task is to follow the generated trajectory in a closed-loop manner. The way how the position controller is working in terms of speed, accuracy, and elasticity, is in our approach directly influenced by the environment of the robot.

During the run, the environment generates a direct feedback

![Fig. 3. The ASGUARD wheel. Five compliant legs are mounted around each hip shaft. The dimensions shown are $a = 18 cm$ and $b = 22 cm$](image)

![Fig. 4. Our architecture of the behaviour-based control using proprioceptive sensor feedback](image)
controlling the speed and the direction of movement for each side of the robot.

Changing the proportional part of the position control parameters on-line has an effect like a mechanical spring. The stiffness at each leg is directly adjustable by the position control parameter. The controller is more compliant for an actuator if the measured torque is higher with respect to the average torque. This does not directly affect the generation of the pattern, only the actual trajectory of each leg. This is comparable to an injury, like a sprained ankle, where humans get a direct negative feedback on the nerves which will result in a more elastic way of walking without changing the motion pattern directly.

The error of the actual trajectory and the target trajectory is then fed back to the motion pattern generator. If the error gets larger than a freely configurable threshold, the pattern of the specific actuator is synchronized with the actual position of the leg. Thus we close the loop between the trajectory and the position controller, which, to our best knowledge, has not yet been achieved with other hybrid legged-wheeled robots. This is an important feature because we do not have to synchronize the left and right side of the legs while climbing a stair manually or by a fixed stairs motion pattern. This task is performed directly by the adaptive controller. When ASGUARD climbs a stair, the first front leg which has contact with the first step will have a higher torque on the actuator. The controller will release the leg and therefore allow a larger position error.

As stated above, we do not have to ensure that the legs along one cross axis are synchronized, i.e., that they have exactly the same pattern phase. An optimal behaviour in climbing a stair would be to keep the tilt angle of the whole robot minimal while all motors have more and less the same power consumption, depending on course of the inclination and the centre of gravity of the system. The controller keeps implicitly the left and right wheel in phase because it is minimizing the torque discrepancy between all legs. This is achieved, for instance, if the two front legs lift the weight of the robot on the stairs at the same time, given that we have the same motor configuration and mechanical friction within the system, and the weight distribution along the body axis. In order to distribute the torque of each actuator, which is directly related to the measured motor current, we use an approach to modify the proportional part of the position controller which is responsible for following the trajectory of the generated pattern (cf. Equation 1).

\[
P_i = (k_i - \left( \text{cur}_i - \frac{\sum \text{cur}}{n} \right) \ast \iota_i) \tag{1}
\]

\(P_i\) refers to the proportional part of the position controller in leg \(i\) and \(\text{cur}_i\) to the measured motor current in ampere for each motor. The constants \(k_i\) and \(\iota_i\) are used to change the stiffness of the controller with respect to the applied torque on the actuators. The term \(\frac{\sum \text{cur}}{n}\) denotes the average current over all \(n\) actuators.

In [1] we already showed that this simple feedback is sufficient for robust stair-climbing. The reader should note that this approach does not directly change the motion pattern in its phase, frequency, or direction, but changes the way the internal position controller changes the actual trajectory by allowing a larger error between the target and the actual trajectory. The difference between the torque of a specific motor (which is proportional to the motor current) with respect to the average torque gives a negative feedback on the controller. This results in a higher elasticity, similar to an adaptive spring. In our approach, a high positive discrepancy in the motor torque results in a higher elasticity in the leg.

In contrast to a stair-climbing behaviour, the robot has to bring as much power as possible to the ground, especially while accelerating. This is best achieved by a strict position controller within our control architecture (cf. Figure 4). For this we use a simple proportional controller (cf. Equation 2) with maximal error amplification.

\[
O_i = P_{\text{max}} \ast \text{error}_i \tag{2}
\]

To make our first control approach (cf. Equation 1) versatile for flat as well as for steep terrain and stairs, we take into account the inclination of the robot which is measured by an inertial-based tilt sensor. We assume that the robot should adapt its control parameters if the inclination is positive, i.e.

\[
0 \leq \frac{\text{pitch}}{90^\circ} \leq 1. \tag{3}
\]

If the inclination is negative, which happens frequently in rough terrains, we set the compliance term to 0, resulting in a non-compliant controller. We extend Equation 1 by applying the measured system inclination, resulting in Equation 4.

\[
O_i = (P_{\text{max}} - (P_{\text{max}} - P_i) \ast \frac{\text{pitch}}{90^\circ}) \ast \text{error}_i \tag{4}
\]

\(O_i\) refers to the motor output of the controller; \(\text{error}_i\) names the difference between the actual and the target position of leg \(i\), respectively.

IV. EXPERIMENTAL RESULTS

The following results were acquired on a standard stair with five steps until the first landing was reached. Note, that for the presentation of our results, only the left side of the robots motor are displayed, because on a stair, without active steering, the left and right actuators produce comparable proprioceptive data and can be neglected for further analysis. The first run on the stairs was performed without any adaptive control, which means that we used a pure position controller to follow the generated leg trajectories for each leg (Figure 5). This resulted in an unstable stair-climbing behaviour because the rear legs were rearing up the robot’s corpus. This again resulted in a critical pitch angle of the robot.
According to the data, the current in the rear motors can easily reach 7 ampere. The position controller follows the generated trajectory pattern with a position error of maximal 50 tics which is equivalent to \( \frac{4}{45} \pi \) (1472 tics \( \equiv 2\pi \)). The synchronisation of the generated pattern with the leg position is disabled (no sync signal is triggered).

On a second trial on the same stairs we used an load/inclination-adaptive controller which implements Equation 4. The adaptive approach leads to a higher compliance of the legs, especially of the rear legs, reducing their torque significantly while climbing the stairs. This is achieved by reducing the proportional part of the position control, allowing a higher position error while following the trajectory of each leg. This is done for each leg individually. Due to the reduction of torque in the rear legs, the front legs have to carry a higher load of the system. This leads to a more balanced torque and power distribution along the body axis. During the run on the stairs, the controller adapted the proportional part of the internal position controller, thus allowing a spring-like, or compliant behaviour, of each of the four actuators (cf. Figure 6). Whenever the error between the trajectory and the current position gets higher than \( \frac{1}{8} \) of the angle between two legs (\( |\frac{1}{40}\pi| \)), the generated pattern is synchronized with the current leg position. The upper row in Figure 6 shows, when a pattern synchronization signal is raised within the system. This has an effect similar to a mechanical friction clutch, but implemented on an electronic level. The advantage of this implementation is that the force, at which the friction clutch is acting, can be defined on-line, even during a run. The friction clutch-like behaviour can be interpreted as follows: When the position error between the trajectory and the current leg position gets larger, the torque also grows proportionally. When the error (and implicitly the motor torque) crosses the threshold, the internal pattern generator is reset, reducing the error to zero. The pattern generator is not stopped at this point, causing the position error to grow again continuously, if the specific leg is blocked or under significantly higher load relative to the other legs. While the blocked or overloaded leg is not acting any more, the other legs are used to push the system forward and are eventually reducing the load of the overloaded leg.

An important effect is that by having this clutch like-effect in combination with the leg compliance, the legs are implicitly left/right synchronized on a stair. Therefore there is no need to select a predefined walking pattern for a stair. The

![Fig. 5. The proprioceptive dataset of the robot climbing a stair without the adaptive controller. A pure position controller is used to follow the pattern trajectory.](image1)

![Fig. 6. The proprioceptive dataset of the robot climbing the same stair with the adaptive controller enabled. A proportional part of the internal position controller is adaptive in relation to the torque and inclination.](image2)
quantitative analysis of the power load distribution is shown in Table I. We found out that by using our adaptive controller, the energy distribution between the front and the rear axis could be reduced from the ratio 1 : 3.6 to 1 : 1.8 during the experiments. Additionally, the maximum load on the rear axis was reduced significantly from 7 ampere peak to roughly 5 ampere peak.

<table>
<thead>
<tr>
<th></th>
<th>Position Controller</th>
<th>Adaptive Controller</th>
</tr>
</thead>
<tbody>
<tr>
<td>Front Legs</td>
<td>2.03 mAh</td>
<td>2.77 mAh</td>
</tr>
<tr>
<td>Rear Legs</td>
<td>7.41 mAh</td>
<td>5.02 mAh</td>
</tr>
<tr>
<td>Power Distribution Front/Rear</td>
<td>1:3.6</td>
<td>1:1.8</td>
</tr>
</tbody>
</table>

**TABLE I**

POWER CONSUMPTION BETWEEN THE FRONT AND REAR LEGS USING A PURE POSITION CONTROLLER (LEFT) AND THE ADAPTIVE CONTROLLER (RIGHT) WHILE CLIMBING THE SAME STAIRS.

V. CONCLUSION AND FUTURE WORK

A. Conclusion

In this work we described a proprioceptive control approach for our hybrid legged-wheeled robot ASGUARD. The robot is controlled by using individual motion pattern generators for each of the four actuated leg wheels. We presented our layered architecture which is using a closed-loop feedback between the pattern and the internal position controller. In contrast to existing hybrid legged-wheeled robots, we did not use a fixed or predefined motion pattern for stairs or flat terrain. The patterns are generated and modified by the direct force feedback applied to each leg. This was achieved by a direct coupling between the applied torques and the stiffness of the position controller. We demonstrated that by only using a proprioceptive torque feedback the robot is able to climb stairs and, when taking into account the tilt measurement of the robot body, is also able to walk fast on flat and levelled terrain.

B. Future Work

Current research is directed to the analysis of different motion patterns with respect to the substrate the robot is moving on while running on flat ground. We suspect that different speeds as well as different types of substrate are an important factor while choosing the type of global motion pattern. This can be observed by many quadruped animals which adapt their walking pattern in regard to locomotion speed and ground substrate.