Control of a Six-Legged Robot Walking on Abrupt Terrain

Enric Celaya and Josep M. Porta
Institut de Cibernètica (UPC – CSIC)
Diagonal 647, 08028-Barcelona, SPAIN

Abstract

Legged robots are well suited to walk on difficult terrains at the expense of requiring complex control systems to walk even on flat surfaces. But simply walking on a flat surface is not worth using a legged robot. It should be assumed that walking on abrupt terrain is the typical situation for a legged robot. With this premise in mind, we have developed a robust controller for a six-legged robot that allows it to walk over difficult terrains in an autonomous way, with a limited use of sensory information (no vision is involved). This walk controller can be driven by an upper level which need not be concerned about the details of foot placement or leg movements, taking care only of high level aspects such as global speed and direction.

1 Introduction

When compared with wheeled or tracked locomotion, legged locomotion is broadly recognized as superior in its capability to traverse irregular and difficult terrains with accidents such as obstacles, cliffs, slants, etc. The large diversity of existing walking animals offers innumerable examples of the possibilities of this form of locomotion. An important drawback of legged machines is the complexity of the control required to achieve effective walking even in completely flat and horizontal surfaces in which much simpler wheeled machines work perfectly well. This means that the use of legged machines is only justified if they can be made to walk on irregular terrain with a certain degree of confidence.

Most approaches to robot walking begin by considering the issue of walking on flat ground as the basic one, solving the problems of stability and gait generation for this case, usually avoiding the use of sensors. Then, the problem of walking on uneven terrain is solved by altering the basic walking routine in order to cope with the obstacles detected using different kinds of sensorial information. Thus, for example, in the OSU Hexapod vehicle, Klein et al. [6] use force and attitude sensors in order to modify a basic control algorithm to permit adaptation to irregular terrain, Ozguner et al. [8] make use of visual information with the same purpose, and Gorinevsky and Shneider [7], working with a small hexapod, compute appropriate corrections to commanded forces and leg positions depending on the soil properties. Making the case more explicit, in an analysis of the design of the robot Attila, Binnard [1] emphasizes the differentiation made between walking (moving at a constant speed over relatively even terrain) and climbing (stop at a large obstacle and search for footholds) as two different tasks with different hardware and software requirements. In agreement with this view, Ferrell [5] designed a control for this robot in which the aspects of walking and climbing are clearly differentiated, providing specific strategies for different kinds of obstacles.

In the present work we take a rather different approach. We assume that the natural habitat of a legged robot consists mainly in irregular or abrupt terrain. Therefore, instead of developing a strategy to walk on flat ground and then introducing some modifications to cope with ground irregularities, we first provide the robot with the capability to stay safely on difficult terrain, trying to keep a stable position even in the case of eventual movements of the ground or of the robot, and only then we address the problem of walking. Flat ground is seen just as a particular case in which, in ideal conditions, a regular gait pattern is likely to emerge, rather than being specifically intended.

Under this perspective we have built a basic control scheme that allows the robot to advance while adapting to ground irregularities, which constitutes a lower level of control that achieves unpurposeful navigation on abrupt terrain, comparable to the level at which a wheeled vehicle would keep its motors running at a constant speed in order to keep the vehicle advancing on flat ground. Above this lower level, a higher level can be built that, depending on its current goals, drives the robot by controlling aspects such as speed and direction.

2 The Robot

Genghis II (IS Robotics) is a commercially available six-legged, autonomous robot. Each leg has two degrees of freedom (dof), *advance* and *lift*, powered by two motors that we call alpha- and beta-motor, respectively. The alpha-motor moves the leg around a vertical axis determining its alpha-position, that is conventionally taken to be zero when the leg is perpendicular to the body. Similarly, the beta-motor moves the leg around a horizontal axis determining its beta-position.

Each motor is provided with a force sensor that works by measuring the current used by the motor. A problem with this is that its value is related with the force only when the leg is still, but legs are not provided with position not velocity sensors, and there is no way to know if a leg is moving or not (since legs are commanded by goal position, all we know is the position the leg is trying to reach, not its actual position). Note also that force readings provide only its absolute value, not its direction.

The robot is equipped with two whiskers that provide obstacle detection, a tactile sensor along the lower side of the body, and a pitch inclinometer (as well as different infrared sensors that we will not consider in the present work.)

The mechanical capabilities of Genghis are somehow restricted: While in normal conditions its motors are powerful enough to sustain the body, in some cases they are not able to raise the body from a low position. On the other hand, legs are unarticulated and relatively short, constraining their feet to relatively small work areas.

3 The Control

We have decomposed the task according to the subsumption architecture [2], in which a number of processes hierarchically organized in layers run independently, communicating between them through message passing. Each new layer with all those bellow it, gives rise to a *level of competence*, that provides the robot with a certain performance. Each layer is composed of one or more behaviors that can include several processes. A behavior can be compared to a "reflex act" or to an "instinct" of the robot: a tendency to react in a certain way in a particular situation.

Next we describe each of the levels in which the task of walking on abrupt terrain has been decomposed (Fig 1).

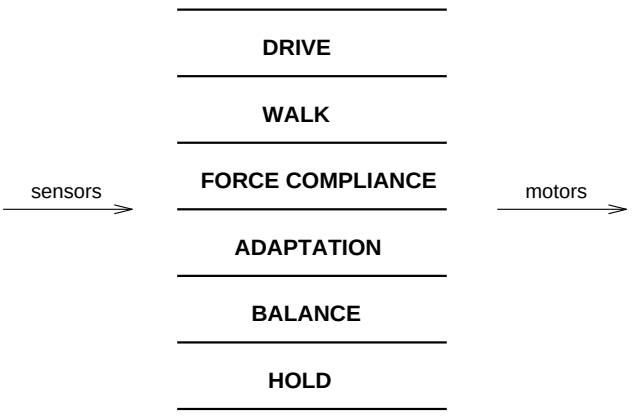

Figure 1: *Level structure of the controller.*

3.1 The Hold Level

The first level is responsible of the direct control of the motors, its main task being to keep each leg at the last commanded position. By default *hold* sets the robot in the *stand-up* position, with all legs oriented in a direction perpendicular to the body, and at an appropriate angle with ground to keep the robot raised, in a trade-off between a decrease of stability when legs are too vertical and an increase of the torques needed to sustain the body when legs are too horizontal. Incoming messages from higher levels can modify the default position of each motor, which can be specified in absolute or relative terms. For each motor, the last commanded position is made available to other levels through an output message.

3.2 The Balance Level

The purpose of this level is to keep a correct body attitude and stance when legs are moved away from the *stand-up* position.

Usually, when a leg is moved by a higher level behavior, what is relevant is its displacement with respect to the other legs, more than its absolute position with respect to the body. Thus, for example, if a leg that has reached its mechanical end of travel is asked to move further in the blocked direction, a possible way to get a similar result is to move all the other legs in the opposite direction. Similarly, if all legs happen to be, say, above or near horizontal, lowering all of them by the same amount should let the robot in a more appropriate stance.

We have devised five different behaviors, that we call *balances*, to control five different aspects of the body's attitude and stance. Each balance works by translating or rotating the body with respect to feet

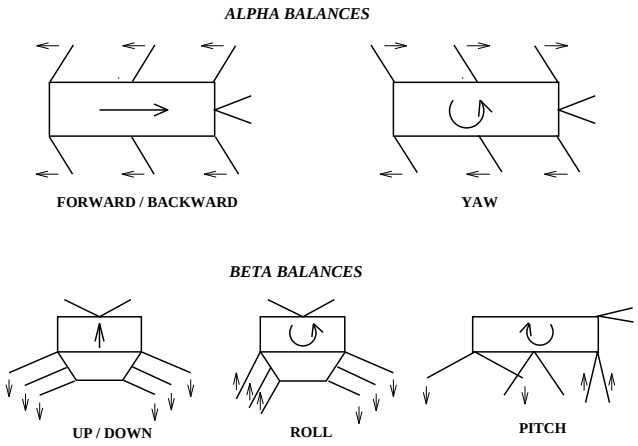

Figure 2: *The five dof of the body controlled by the balance level.*

according to one dof (Fig. 2). Ideally, each of the balances should move the body while maintaining all feet in their original positions. Unfortunately, this is not possible in Genghis, mainly due to the fact that its legs are unarticulated. That this is so can be readily seen when comparing the problem with that of positioning a generalized Stewart platform. It is well known that, except for some special configurations, the platform cannot move if the lengths of the six links attaching it to ground are fixed. In our case links correspond to legs, whose length is constant, implying that any body movement requires a change in the relative positions of feet. However, the actual feet displacements are relatively small, and we will neglect them.

The five balances are described next.

1. *Global Alpha Balance*

   Controls the advance position of legs by moving all of them by the same amount in order to keep their average alpha-position near zero. Its net effect when legs are on ground is a translation of the body along its longitudinal direction.

2. *Lateral Alpha Balance*

   Keeps the average advance position of right legs near the average advance position of left legs by issuing the same alpha-displacement to all right legs, and an equal but opposite alpha-displacement to all left legs. Its effect on the body is a rotation about a vertical axis (yaw) in the direction of the side with less advanced legs.

3. *Global Beta Balance*

   Controls the lift position of legs by raising or lowering all of them by the same amount in order to keep their average beta-position near the default value used in the *stand-up* position. Its net effect is a vertical translation of the body.

4. *Lateral Beta Balance*

   Keeps the average lift position of right legs near the average lift position of left legs by issuing the same beta-displacement to all right legs, and an equal but opposite beta-displacement to all left legs. Its effect on the body is a rotation about a longitudinal axis (roll) in the direction of the side with less raised legs.

5. *Frontal Beta Balance*

   Keeps the average lift position of the two front legs near the average lift position of the two rear legs by issuing the same beta-displacement to both front legs and an equal but opposite beta-displacement to both rear legs. Its effect on the body is a rotation about a transversal axis (pitch) in the direction of the end with less raised legs.

An important property of this set of balances is that they are orthogonal, i.e., the action of one of them has absolutely no effect on none of the others, and each balance can be adjusted independently of the actions of the others.

As we have seen, the five balances allow to drive the body along five of its six dof. This begs the question of why a sixth balance has not been introduced. As can be seen in Fig. 2, the missing dof corresponds to lateral translations. A little thought will convince the reader that, when the robot is in the *stand-up* position, there is no possible combination of leg movements that produce a lateral translation of the body[1]. Fig. 3 shows how, by leading the robot to an appropriate stance, in which front and rear legs are nearly aligned with the body, a lateral translation movement can be obtained. Clearly, such a position of legs is not appropriate for normal walking, and would be unfeasible to adopt it each time the corresponding balance had to work. This fact prevented us from introducing the sixth balance. This problem is not present in other robot structures, as for example in those with articulated legs. In these cases the missing balance should be introduced. Note however that if our robot is not able to act on the sixth balance, it is neither able to corrupt it, then it will always be balanced and its correction is unnecessary.

[1] Strictly speaking, the action of *lateral beta balance* slightly translates the body sideways at the same time it rolls it: both dof are linked and it is not possible to drive them independently. We dismiss the effect of this lateral translation since in general it is less important than that of the rotation.

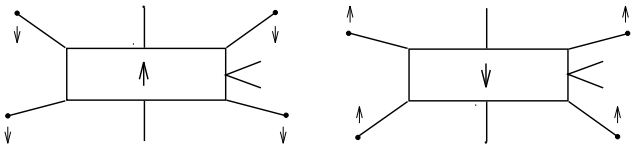

Figure 3: *Leg positionings for lateral displacements. Middle legs should not touch ground or they will slip on it.*

3.3 The Adaptation Level

Though the balanced positions achieved by the previous level are adequate when the robot lies on a horizontal surface without disturbing obstacles, in more conflictive situations they can be non-optimal. The purpose of the adaptation level is to change the targets aimed at by the different balances in order to better fit the environmental conditions detected through sensors. This level is composed of three behaviors:

1. *Advance Adjustment*

   Modifies the target average alpha-position of legs controlled by *global alpha balance*, which by default is taken to be zero. Its action responds to the pitch detected by the inclinometer, advancing or backing the body depending on the angle formed with the horizontal, in such a way that legs adopt a more vertical position, and the robot's center of mass approaches the center of its supporting polygon, thus increasing its stability.

2. *Height Adjustment*

   Modifies the target average beta-position of legs controlled by *global beta balance*, which by default is taken to be that of the *stand-up* position. This behavior increases the elevation of the body when the touch sensor on the lower part of the body has registered a recent contact.

3. *Attitude Adjustment*

   This behavior modifies the action of the two rotational beta balances, *lateral beta* and *frontal beta*, producing a different body attitude. Its action is determined by the recent contacts detected by the whiskers. The way it works is by "fooling" the beta balances by artificially incrementing, for the leg nearest to the whisker, the beta-position value to be used in the balance computations. Its effect is a tendency to raise the body at the side where the contact is detected, in an intent to overpass the obstacle that produced it.

A behavior to adjust the *lateral alpha* balance controlling the body's yaw has not been added, though, in fact, it was implemented as a third component of the *attitude adjustment* behavior, but finally we considered it unnecessary and removed it.

Finally note that if the robot had a sixth balance allowing sideways translations (which would be possible in other robots), a *lateral adjustment* behavior could have been implemented to control the lateral displacement of the body. In this case, the sensorial information governing this behavior would be the roll inclination of the body, which in our robot can not be detected due to the lack of the corresponding sensor.

3.4 The Force Compliance Level

At this level, the robot adapts the height of each foot to the elevation of the ground just below it. This assures that all feet are lying on the ground, supporting part of the robot's weight and increasing its stability.

The way this is achieved consists in introducing, for each leg, a behavior that continuously monitorizes the force supported by its beta-motor. When this force is found to be under a given threshold, the leg is lowered a bit (ignoring the possibly interfering orders coming from the *balance* level.) This strategy, when combined with the effect of *balance*, constitutes an effective form of active compliance. Observe that there is no explicit behavior to raise an overloaded leg, but that the same effect is indirectly obtained from the interaction between the *beta balance* and *force compliance* behaviors: If some legs are overloaded, it must be the case that other legs are underloaded, which, consequently, will be lowered by the corresponding *force compliance* behaviors. The *global beta balance* behavior will then correct the situation by raising the overloaded legs.

In its work with a previous version of Genghis, Brooks [3] introduced a force balancing behavior that consisted in backing off a leg whose force raised beyond some threshold when placed on the ground. A problem found with this approach is that, in high pitch situations, the rear or front legs tend to be overloaded, what causes them to be raised increasing the pitch even more. Brooks tried to solve this problem by monitorizing the pitch inclinometer and inhibiting the force balancing behavior in high pitch situations. This problem never appears in our approach since *frontal beta balance* avoids such situations without the need of any pitch information. The problem was observed, however, when *frontal beta balance* was removed in an experimental test.

A potential problem with the active compliance scheme used here is that of oscillations around the

equilibrium point. We solved this problem by a simple strategy consisting in triggering the balance behaviors only when the corresponding deviation is beyond a given threshold.

3.5 The Walk Level

At this level the robot is able to walk on abrupt terrain, trying always to move forwards crossing over any obstacles it finds on its path.

The structure of this level is simple: For each leg, there is one *step* and one *skip* behavior. The *step* behavior waits for messages from its two neighbor legs (Fig. 4) announcing that they have already issued its stepping movement. Then, as soon as both neighbor feet are simultaneously in contact with ground, the stepping movement, consisting in raising and advancing the leg, is issued, and the corresponding signal to the neighbor legs is sent. After initializing the system with an appropriate set of messages, the sequence of mutually triggering steps is self-sustained. This strategy guarantees that each leg issues exactly one step between two consecutive steps of a neighbor leg and, at the same time, that never two neighbor legs are stepping at the same time. The advance of the robot is produced by the backward movement of legs that are on ground ordered by *global alpha balance* each time a leg disrupts this balance by a forward stroke.

Since the time a leg should wait between steps depends on the time their neighbors take to complete a step and reach the ground, the resulting gait is not previsible in advance and depends on the terrain conditions. Note, however, that in a completely flat ground, it is likely that all legs take the same time to complete its cycle, and, as can be observed in the real robot, the leg sequence tends to the tripod gait, which is known to be the fastest and most efficient statically stable gait. As terrain conditions become harder, synchronization between legs is lost, gait is altered and the effective speed is automatically reduced, being at any time as fast as circumstances permit.

The second behavior, *skip*, is awakened when a force is detected in the alpha-motor while the leg is performing its stepping forward movement, what means that a collision with an obstacle has occurred. In this case *skip* moves the leg back a short distance and issues a new stepping movement with a slightly increased elevation. This action is repeated as many times as an alpha-force is detected.

3.6 The Drive Level

This level implements a rudimentary avoiding behavior consisting in driving the robot away from the obstacles detected by whiskers. Our strategy is the widely used "turn away from the obstacle".

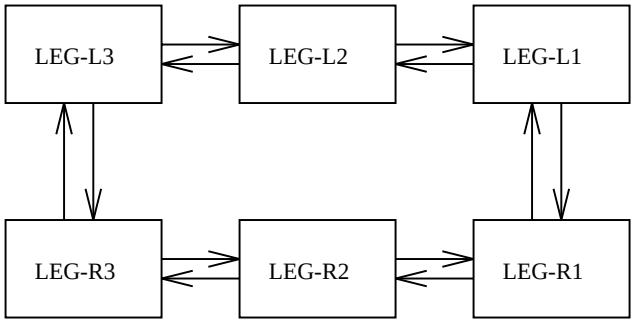

Figure 4: *Neighbor relationships between legs.*

The *drive* layer has only two behaviors, one for each side, controlling the stroke (step's length) of the legs on each side according to the recent history of contacts detected by the whisker of the opposite side, progressively shortening it when many contacts are detected and enlarging it when not.

The turn is obtained through the effect of *lateral alpha balance*, in a very similar way as *global alpha balance* produces the advance of the robot. What produces the turn is the difference in the compensating movements of legs in the two sides of the robot, that are required to compensate the different stroke lengths of the legs in both sides. Our approach differs from the more usual one ([3], [4], [5]) consisting in limiting the travel of legs by placing a stopping point. In this case, each leg moves until its limit position is reached and stops while other legs continue to move, thus forcing the leg to be dragged along the ground.

Note that, in our approach, the speed of the robot depends only on the average length of the strokes, and that simply reversing the direction of the strokes, the robot walks backwards. This allows to control the speed and direction of the robot easily and continuously. The turning speed can also be varied by simply modifying the difference between the average stroke lengths on both sides of the robot. If strokes on both sides are equal and opposite in direction, the robot will turn in place. Note that advance and turning speeds are independent and can be controlled separately.

4 General Performance

A quantitative measure of the robot's performance is given by the maximum height it is able to climb. Using our controller, the robot is able to cross over vertical steps of 11cm, which roughly corresponds to the leg length, and is notably larger than the 8cm of

clearance existing between the body and the ground when legs are in vertical position. For the sake of comparison, using the program "force-walk" provided with Genghis as a demo, the robot is unable to cross over steps of more than 5cm.

To evaluate the general performance of our controller, we have made extensive tests on artificial landscapes made of wooden blocks, with irregularly distributed slants and steps of sizes ranging from 1 to 10cm. Many aspects of the general performance, such as smoothness of walking, accommodation of feet to ground, body attitude with respect to the local support surface, etc., have been greatly improved, though they are difficult to quantify.

An unintended emergent effect has been observed which consists in a swing of the body from one side to the other in the process of walking. This effect is due to the action of *lateral alpha balance* that reacts very fast to the temporary differences generated by successive strides, causing a rotation (yaw) of the robot at each step. The swinging is not pernicious but, if wanted, can be eliminated by allowing a margin of tolerance in *lateral alpha balance* to avoid its action when differences are small.

A related and more interesting effect appears when a leg is raised to reach a high position. In this case the balance behaviors react by lowering the other legs in such a way that the body of the robot is raised, helping the first leg to reach even higher. A completely analog effect occurs in a horizontal direction when a leg is moved forward.

5 Conclusions

A hierarchical control for a six-legged robot has been built that allows it to walk on abrupt terrain. The procedures and general organization in layers described here are applicable not only to our particular robot, but to any robot that performs statically stable walking. With some adaptations the controller could be used in robots with articulated legs, or even in robots with four or eight legs.

The *walk* level can be seen as a low level of locomotion that keeps the robot advancing on rough terrain, comparable to the level at which a wheeled vehicle keeps the robot advancing on flat ground by turning its wheels at a constant speed. The *drive* level constitutes an example of an upper level that controls *walk*, driving the robot according to its own navigational purposes.

A major contribution of this paper is the use of a set of orthogonal balances to drive the body of the robot along different dof while feet stay on ground. This approach has proved to be very fruitful, and lead to interesting ways to achieve stability, force-compliance and velocity control.

Finally, we want to emphasize the fact that a robot must be able to stay safely on rough ground before trying to walk on it, a principle that, even if it may seem a trivial truth, has been disregarded in many previous approaches to robot walking.

Acknowledgments

This work has been partially supported by the Comisión Interministerial de Ciencia y Tecnología (CICYT), under the project "Subsymbolic techniques for constraint satisfaction, vision and robotics" (TAP93-0451) and by a grant from the Comissionat per a Universitats i Recerca de la Generalitat de Catalunya to the second author.

References

[1] Binnard, M.B. (1992): "Leg design for a small walking Robot", B.Sc. Thesis, MIT.

[2] Brooks, R.A. (1986): "A Robust Layered Control System for a Mobile Robot", *IEEE Journal of Robotics and Automation*, vol. RA-2, No. 1, pp. 14-23, March.

[3] Brooks, R.A. (1989): "A Robot that Walks; Emergent Behaviors from a Carefully Evolved Network", *Neural Computation*, No. 1, pp. 253-262.

[4] Chiel, H.J., Beer, R.D., Quinn, R.D. and Espenschield, K.S. (1992): "Robustness of a Distributed Neural Network Controller for Locomotion in a Hexapod Robot", *IEEE Trans. Robotics and Automation*, V. 8, N. 3, June 1992, pp. 293-303.

[5] Ferrell C. (1993): "Robust Agent Control of an Autonomous Robot with Many Sensors and Actuators", M.S. Thesis, MIT.

[6] Klein, C.A., Olson, K.W. and Pugh, D.R. (1983): "Use of Force and Attitude Sensors for Locomotion of a Legged Vehicle over Irregular Terrain", *Int. Journal Robotics Research*, V. 2, N. 2, pp. 3-17.

[7] Gorinevsky, D.M. and Shneider, A.Y. (1990): "Force Control in Locomotion of Legged Vehicles over Rigid and Soft Surfaces", *Int. Journal Robotics Research*, Vol. 9, No. 2, pp. 4-22.

[8] Ozguner, F., Tsai, S.J. and McGhee, R.B. (1984): "An Approach to the Use of Terrain-Preview Information in Rough-Terrain Locomotion by a Hexapod Walking Machine", *Int. Journal of Robotics Research*, Vol. 3, No. 2, pp. 134-146.