Model-Free Active Balancing for Humanoid Robots

Sara McGrath[1], John Anderson[2], and Jacky Baltes[2]

[1] Cogmation, Winnipeg, Canada; `ummcgrath@cs.umanitoba.ca`
[2] Autonomous Agents Lab, University of Manitoba, Winnipeg, MB, Canada,
R3T 2N2; `andersj,jacky@cs.umanitoba.ca`; http://aalab.cs.umanitoba.ca

Abstract. To be practical, humanoid robots must be able to manoeuvre over a variety of flat and uneven terrains, at different speeds and with varying gaits and motions. This paper describes three balancing-reflex algorithms (threshold control, PID control, and hybrid control) that were implemented on a real 8 DOF small humanoid robot equipped with a two-axis accelerometer sensor to study the capabilities and limitations of various balancing algorithms when combined with a single sensor. We term this approach a *model-free* approach, since it does not require a mathematical model of the underlying robot. Instead the controller attempts to recreate successful previous motions (so-called baseline motions). In our extensive tests, the basic threshold algorithm proves the most effective overall. All algorithms are able to balance for simple tasks, but as the balancing required becomes more complex (e.g. controlling multiple joints over uneven terrain), the need for more sophisticated algorithms becomes apparent.

1 Introduction

For humanoid robots to move from fantasy to reality and become practical, they must be able to move over a variety of different terrain with different speeds and gaits without falling over and without the need to manually tune the gait. At present few small humanoid robots competing at the large international robotics competitions FIRA and RoboCup use any type of feedback control, and do little in the way of active balancing. For example, the technical challenge at RoboCup 2005 involved a section of uneven terrain, but this challenge was removed because most teams were unable to complete it. Of the few that did, some effectively cheated by adding skis to the feet of their robot.

Integrating dynamic balancing into robots will allow them to not only deal with changing surfaces, but also allow them to compensate for sudden changes in their equilibrium. Further, balancing will also allow robots to move to new gaits and tasks, such as crawling or load-bearing with greater ease and robustness.

Humans themselves use multiple sensors to balance: vision, position/force feedback (muscle feedback) and tilt/acceleration sensors (inner ear organs). Based on an analysis of the dynamics of a robot, force feedback and motion based

sensors have been used by many researchers in combination. A lot of these approaches have only been implemented in simulation and require an accurate mathematical model of the robot.

Most human balancing occurs subconsciously using balancing reflexes that can be demonstrated (by, for example tapping a human on the shin). These reflexes are present in any human motion, allowing people to preform basic balancing whatever their actions may be. Little research has been done on simple balancing reflexes for humanoid robots that employ a simple algorithm and a single sensor. However, to evaluate the capabilities and limitations of various algorithm and sensor combinations, this type of experimentation is extremely important. Multiple sensors potentially mask or ignore important sensor data.

To develop balancing reflexes, we decided to investigate each sensor individually and determine the extent of its usefulness.

A two-axis accelerometer was mounted on *Lillian*, a humanoid robot, as the sole balancing sensor. The choice of an accelerometer was based on the fact that these are small, cheap, and easily available. We often use mobile phones for on-board computation in our robots, and many modern mobile phones come equipped with accelerometers, but not other sensors. Another reason was that accelerometers are similar to the human inner ear and can thus be used to study and exploit human balancing reflexes.

Lillian, a robot from the University of Manitoba's Autonomous Agents Lab, has 8 degrees of freedom (DOF) (actuated with servo motors), a Memsic 2125 two-axis +/- 1.5g accelerometer and an Eyebot controller board. Figure 1 shows a model of the actuator setup for Lillian. Lillian uses 8 DOF, 4 for each leg. The ankle can be moved in the sagittal plane using joints labeled XAS as well as the frontal plane using the joints labeled Y. One degree of freedom in the knee allows movement in the sagittal plane using joints XK. The hip can also be moved in the sagittal plane using joints XH.

The main design principles in building the robot were simplicity and frugality, as this forces the resulting algorithms to be more robust, versatile solutions.

2 Related Work

Several special-purpose algorithms to adjust a robot's motions are described in the literature. They can broadly be classified into two types: (a) Center Of Mass (COM) based algorithms keep the robot's COM in the supporting polygon of the robot's feet, (b) Zero Moment Point (ZMP) algorithms calculate the point in the horizontal plane at which all the moments are zero, and keep the ZMP in the supporting polygon. On the other hand, the versatile PID (Proportional Integral Derivative) controller is a basic control strategy that adjusts the error of a feedback output to a desired reading (baseline), making corrections based on a percentage of the error, and the integral and derivative components of the error. Similarly threshold control, with minimal corrections applied to readings outside a given threshold, was developed in previous research by [1].
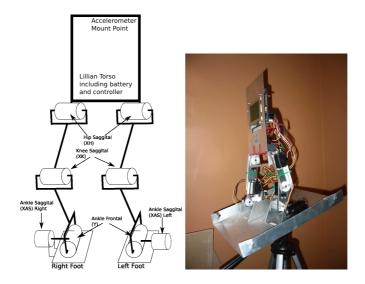
Fig. 1. Lillian, an 8 DOF robot. Kinematics are shown on the left, the actual robot during a tilting test with mounted accelerometers on the right.

Control methods can also be divided into simple balancing reflexes, with tight control loops between sensors and simple algorithms, and more specialized algorithms such as AutoBalancer [2]. Exploration into new sensors tends to begin with simpler control methods and then more complicated algorithms. While these intensive algorithms can use more complicated methods to be fine-tuned (ie, reinforcement learning, genetic algorithms), they apply to the specific robot, and are more computationally intensive to create, often requiring preprocessing. Reflexes use less complicated methods, and are simpler, more general functions that could ideally transfer more effectively than do the highly-tuned algorithms.

Huang et al. [3] investigated using sensory reflexes on their robot BHR-01, incorporating a ZMP reflex, a landing phase reflex, and a posture reflex into the dynamic walking pattern. These reflexes were triggered by sensory information, and when active, would compensate for any imbalances in the walk by adjusting the ankles, hips or knees. The corrections were used to adjust the offline pre-calculated walk pattern. These realtime reflexes, added to the walking pattern, proved effective in walking over uneven terrain.

Team KMUTT [4] used a simpler velocity based control to dynamically balance their robot. This balancing mechanism is part of a specific walk, chosen by the robot if its sensors indicate conditions are appropriate, not a standard part of the robot's behaviour. The robot has two PD-controlled walks: a slower static walk that uses the force sensors on the robot, and a faster dynamic walk that balances using accelerometers and gyroscopes. In the static walk, the PD controller manipulates the height at the robot's hip based on the force sensors in the

foot. The dynamic walk controls the velocity at the hip with its PD controller. Team KMUTT competed at RoboCup 2006 using this code.

The University of Manitoba's Tao-Pie-Pie is the sole robot to use only gyroscope readings for active balancing [1]. The readings are processed and run through a Threshold controller, compensating for perturbations in the gait. The Threshold controller simply applies a minimal correction when sensor readings break a predefined boundary. Balance is explicitly added-on as corrections made to the pre-calculated walk gait. These corrections were used in competition to compensate for the poor surface, and were found to be better than the previous gait. Further, the Threshold method is extremely simple to implement and tune.

3   Determination of the Baseline Walk

Most approaches to humanoid balancing are model-based. In a model-based approach, the researcher creates a mathematical model of the robot to work on. Then a control algorithm is implemented and tested on the model (often in simulation). Finally, the controller is moved to the physical robot. Often, significant readjustments are required at this point to transfer the controller. One of the major disadvantages of this method is the fact that it is often difficult to develop a mathematical model for an existing robot that is accurate enough to help develop balancing algorithms.

Model-based control approaches attempt to overcome this problem by adding sensors to implement a feedback loop to overcome limitations in the model or external disturbances. This is problematic as in many cases the control variable of interest (e.g., center of mass or ZMP) can not be measured directly. So a controller will only be able to provide feedback via indirect measurements (e.g., joint positions). Unfortunately, the mapping of joint positions to ZMP is dependent on the model and the robot will fall over even if small changes to the weight of arms or torso are being made.

In contrast, our approach is implementation-based and does away with the need for a model of the robot. The robot itself is used as a test platform. This removes the necessity of adjusting the control strategy for the robot or unforeseen physical factors, as the testbed is perfectly accurate. The drawback is in maintaining the robot, as well as the length of time required for each test. Our methodology is to modify a pre-existing gait to improve it. Thus, our control algorithms are applied to the current gait, allowing the results of each strategy to be directly compared with each other and the uncorrected walk on the real robot.

Our approach relies on being able to record baseline motions (previously successful motions and their associated sensor readings) and on a common sense analysis of the joint motions of the robot (e.g., the position of the robot in the frontal plane is controlled by the ankle and hip servo, which has to compensate for gravity as well as the weight of the torso).

Figure 2 shows the X and Y readings for several dozen steps of a successful walk. As can be seen, most readings in a successful walk fall in a small band.

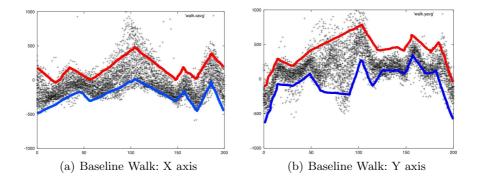(a) Baseline Walk: X axis (b) Baseline Walk: Y axis

Fig. 2. Sensor reading for the baseline walk in the X and Y axis

After analyzing the baseline walk, an upper and lower threshold limit was introduced as shown in Fig. 2. These threshold limits were used in the threshold controller as described in Section 4.

4   Methodology

Three popular control algorithms were used here to convert sensory readings into motion corrections and thus implement a balancing reflex: a standard PID controller, a threshold based controller, and a hybrid version of the two. For an accelerometer-based PID control, a baseline is created by either taking a sample from previously programmed good motions (e.g., a walk gait), or setting the baseline to be unmoving (stand). The closer the baseline conforms to the actual readings, the better the corrections. Threshold balancing first determines a threshold area, where no corrections will be made. This threshold is currently centered on the PID baseline, but with a broader range to avoids corrections from causing oscillation when the robot is standing still. Corrections are only applied once they pass a certain error value. This allows for a simpler means of adjusting settings, and eventual comparison or combination with PID settings. Both methods listed above have their own faults: PID, a quick reaction but a tendency to overcorrect, and Thresholds less of a tendency to overcorrect, but also less able to react quickly to larger errors. Thus a hybrid method was proposed and implemented to combine the best of both methods. It uses Thresholds for smaller corrections, but PID-based corrections for larger errors.

The Sum of Absolute Error (SAE) is a quantitative measurement used here to determine the relative goodness of varying walks. As all the correction methods have a baseline (or a set of thresholds), the deviations can be measured to directly compare trials. Summing the absolute errors measures the total deviations from the baseline. The greater the total deviation, the less the walk conforms to the baseline, and the less the corrections are helping.

As with most balancing approaches, Lillian was tuned for one plane before complicating matters with multiple planes, inclines, or uneven terrain. Further, as differing gait disturbances produce oscillation at differing points in the tuning, small increments of complexity are necessary to allow for the robot to be properly tuned. As many researchers use only one plane, or do not specify a method by which to tune for increasingly complicated balancing, one is given here.

Tuning begins with standing still, before moving to tilting, walking, and then multiple joints and planes. First, the corrective methods are tuned not to oscillate during stands. This gives a minimal value to use as a base for the PID settings and threshold bounds. The next step is for the robot to stand still on a surface that tilts in either the frontal or sagittal plane, thus forcing corrections for deviations from its desired pattern. In order to minimize possible complicating factors, the robot will correct in one axis at a time, attempting to maintain a sensor reading of zero. Once Lillian can remain stable while tilting, a more complicated sensor pattern can be used, and the tuning adjusted. Following a walk baseline on a flat surface instead of a steady line adds another level of complexity to the balancing, exposing previously hidden oscillation, as the balancing must compensate at many levels and speeds. Balancing in two axes is much more difficult than any of the previous tasks, as any oscillation (or even a too quick correction) in one axis can produce a rebound and perhaps oscillation in the other axis. Thus, two axes are not tested until balancing is working effectively in one axis. Again, two axes balancing starts with a simple tilting platform before moving to a more complicated pattern — walking on a even surface, then uneven terrain. Tuning is limited here to walking on a flat surface. Further tests are used to evaluate the algorithms.

While this methodology only uses a walk gait for its most complicated sensor pattern, any gait will have a repetitive pattern that can be used to calibrate a baseline. These gaits include crawling and running, to begin with. Once Lillian has been tuned for multiple planes of balancing, it is possible to replace the walk baseline with one created from a crawling gait, for instance. Corrections would then be made based on the new gait, but use the same tuning as the previous gaits. This allows the balancing reflexes implemented here to transfer quickly and easily to new gait patterns.

Tuning configuration began with the PID and threshold methods on the tilting platform. The test results were used to refine the PID and Threshold walking tests. The best results were used to test the hybrid method. A side-by-side comparison of all the best results was used to choose settings for the final tests with the perturbed walks, and the stepping field tests.

Testing on the tilting platform began with the PID and Threshold correction methods applied to individual joints: ankle sagittal (XAS), knee sagittal (XK), and ankle lateral (Y), and then the best results were used to create further walking tests. The platform was tilted from -30° to +30°, from a starting position of 0°, with an angular velocity of 240° per minute. The tests were coarse grained, running a trial with controller gains of 150, 450, 750 and 1050, with delays of 10, 40, and 70 milliseconds required between corrections.

After tuning the P settings, the best setting from each joint was taken and tested with a range of differing D settings, by setting a baseline that stayed at +1000 for the first half of the gait, and then moved to -1000 for the second half. The only movement was provided by the corrections adjusting the accelerometer readings. It was thus possible to look for the overshoot caused (or avoided) by varying the D parameter, running each trial once for an initial exploration.

Results from the single joint tests were used to tune for correcting multiple joints simultaneously. Results that improved on the baseline where possible, and the best results available otherwise, were selected for further tuning and divided into two (best and good). Each of the best joint settings was paired with each of the best and good settings for another joint, to allow reasonably thorough testing without factorial explosion. Each of these tests were run three times, to reduce the noise in the data. Further, X joints were first tested and combined with each other (XAS and XK) before adding in a second plane with the Y ankle joint. This follows the ideology of the prior tests in adding in as little complexity as possible to each test for a clear picture of the effects of each new factor.

5   Evaluation

A basic walking gait was first used to evaluate the best of the tuning results side by side. Any result that improved on the gait (in the plane it was correcting) was used for the final evaluation. Tests include randomly perturbing the walk to varying degrees, and running Lillian over a stepping field, with and without balancing reflexes, for approximately twenty seconds. This stepping field (Figure 3) was constructed of layered pieces of cardboard, always providing a height difference of 3 mm between neighbouring pieces, but possibly more than one piece over the length or width of Lillian's foot.

Fig. 3. Stepping field used in this evaluation.

Gaits were perturbed by randomly varying the control points of one good gait over a spread of 5 or 10 set points ([-2 .. 2], or [-5 .. 5]), at multiple points

throughout the gait. The disturbances were applied to both joints, as the balancing control assumes that the movements of the joints are coupled.

The sheer number of tests carried out to tune the controllers on the robot make it difficult to determine what settings best improves the walk. Therefore, the best settings in each method, PID and Threshold, were directly compared against each other, though the SAE is normally calculated differently for both. As the thresholds are currently set equidistantly from the PID baseline, the SAE was calculated using the PID baseline on all non-tuning trials.

The PID controller settings chosen were the best for each joint, the best combination of settings for the X joints, the best for all the joints, and the best P and PD controllers for Y. No D settings were chosen for any of the X joints as they had not improved the controllers. found to improve on the basic walk, no further combinations of settings involving them were chosen.

The Threshold settings were similarly chosen as the best setting for each joint, and the best combinations of X and XY joints. Two XY settings were chosen, as one improved on the basic walk for Y, and the other was the best set of corrections for X. Two more settings were chosen for testing: XAS + Y and XK + Y, as one of the XY settings was the same as the three best individual joint settings. Further, each of the single joint settings (unlike the PID controller) actually improved on the basic walk, suggesting that a single X joint setting with a Y setting could prove effective.

As the hybrid controller depends on combining the Threshold and PID controllers, it was only tested after the best settings for the simpler controllers were determined. The hybrid controller is tested on settings that improve the walk: on the top two XAS settings for both PID and Threshold; the combination of the Y controllers; and for comparison's sake, the two best Threshold XY controllers with the best PID XY controller.
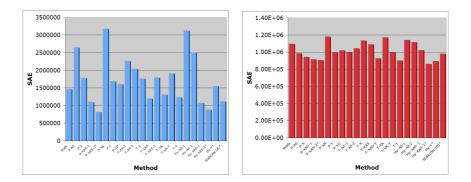

Fig. 4. Random walking test results, by method and settings. Y axis shows SAE of raw sensor data.

Figure 4 shows the results of the direct comparison. Less than half actually improve upon the uncorrected walk, in the plane(s) that they are correcting for. This criteria is used to select settings for the final evaluations, giving P-XAS-1, P-XAS-2, Hy-XAS-1, Hy-XAS-2, T-XAS, T-XK, T-Y, P-Y, and Hy-Y.

5.1 Random Walks

We tested our algorithm by perturbing the walk (simulating control error in the servo motors) by varying degrees (5%,10%, and 15%). Even with small perturbations of 5%, the need for corrections becomes apparent. The SAE of the methods correcting in the X plane is a third to a quarter of the uncorrected methods. Solely Y corrections are not as impressive, leaving the walk unimproved. The differences between correction methods in this test are much less than the difference between the corrected and uncorrected tests.
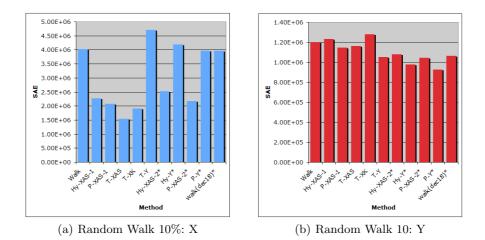

(a) Random Walk 10%: X   (b) Random Walk 10: Y

Fig. 5. Random walking test results with a perturbation spread of 10%, by method and settings. Y axis shows SAE of raw sensor data.

Increasing the perturbations to up to a spread of 10% caused the differences between the methods to become yet more pronounced, as shown in Figure 5. The Threshold results are the best in the X plane, but with a greater difference between them and the other methods. Any correction method still shows a marked improvement over the uncorrected walk.

5.2 Stepping Field

Results from the stepping field (see Figure 6) show that the balancing degrades here. For the X plane, only the PID corrections improve on the uncorrected walk.
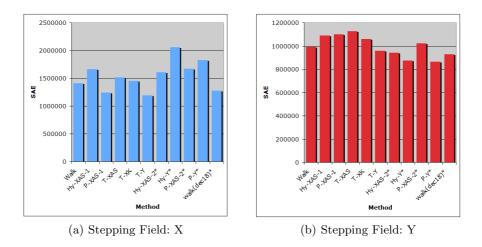
(a) Stepping Field: X   (b) Stepping Field: Y

Fig. 6. Stepping field test results, by method and settings. Y axis shows SAE of raw sensor data.

While the hybrid method actually worsens the SAE readings, Threshold corrections merely leave the walk mostly unchanged. Unlike previous tests, Threshold corrections on the XK joints are actually slightly better than those of the XAS joint. Compared to the differences between the corrected and uncorrected walks in the perturbation test, however, the differences between the results are relatively minor, indicating the X corrections are not making a large difference to the walk. As expected, the lack of corrections to the Y plane by these methods is clearly shown, as the Y readings for all the X correction methods are worse than the uncorrected Y walk. The threshold corrections for Y improve on the uncorrected walk for both X and Y, not just Y. Therefore, unlike the perturbation test, where the main corrections to be made were to the X plane, here the most effective corrections are to the Y plane, and correcting in X without correcting Y will not be very effective.

5.3 Analysis

Directly comparing all results with each other led to some interesting observations. The more joints a controller attempted to control, the worse the balancing became. Similarly, more complicated controllers, such as PD instead of P, did not improve the balancing. The simplest ideas, such as the threshold controller, were just as effective in controlling the walk, and maintaining the desired accelerometer readings. The simpler threshold controller also responded better to adding more joints, not overreacting as much as the PID controller.

This conclusion of complexity not being handled well was further born out by the random walks. Different joints were also more effective in controlling the balancing. The XAS joint is much more effective at controlling the robot's bal-

ance than any of the other tested joints. Further, as the perturbations increased, Threshold showed as more effective than PID. The hybrid controller is still better than no controller, but not as good as either of the simpler controllers.

The stepping field upsets the previous trends with PID actually outperforming Thresholds. This suggests the threshold controller is unable to react quickly and strongly enough for the corrections required. The random walks had a change of perhaps 5 or 10 servo settings over 1 second; the threshold controller allows approximately 10-12 servo setting changes per second. Thus the threshold controller could compensate for the changes in the walk, while the PID controller tended to overreact. Once on the stepping field however, the changes would occur instantly, and require compensation of 5-10 servo settings (or perhaps more). The PID controller could react instantly, while the threshold control would move more slowly to correct. This slower reaction time would allow the robot enough time to accumulate inertia in the wrong direction, making it even more difficult for the robot to compensate. A further explanation of the poorer results on the stepping field is due to the initial assumption that all joint actions are coupled. This is generally true of the testing, as with the randomized perturbations and the tilting, the robot's feet remain relatively aligned with each other. The stepping field, however, due to the unevenness of the terrain, allows for the feet to become misaligned, and this is not easily corrected by the balancing reflex, as currently implemented.

6 Conclusions and Future Work

This research showed that simple reflexes can be used to balance a robot in simple situations, but that they become unable to handle the complexities of more normal situations (ie, walking while controlling multiple joints). Both Threshold and PID algorithms showed impressive results on the tilting tests, with a broad range of settings providing beneficial corrections to the robot. Moving those corrections to a walk demonstrated the shortcomings of the reflex algorithms, as the speed and complexity of balancing a walking robot started to overcome the balancing capabilities of the controllers. This was most noticeable with PID, as Thresholds still had several useful settings. Adding multiple joints to be controlled, or moving the robot to an uneven surface further demonstrated the inability of the balancing reflexes to compensate for the amount of variability in the walk. While the reflexes were not able to fully compensate for any surface, they did improve the walk noticeably against smaller, more regular changes, as shown by the random perturbation tests.

No one algorithm was consistently best; rather, the most effective algorithm depended on the circumstances in which the robot was placed. Overall, Thresholds appeared best for slower, steadier changes, while PID responded better to occasional larger changes. The Hybrid method was never the best method, but was almost always in between the two other algorithms in terms of goodness; never the best, but regularly the runner-up. In every test, however, at least one of

the correction methods matched or outperformed the uncorrected walk, showing that a tuned correction method is better than no correction.

Overall, due to its ease of tuning and general performance, the threshold method is the easiest and most useful choice for future balancing. While there are differences between the methods, for a single joint they all do improve upon the uncorrected walk, and thus any method is better than none.

This research has focused on a walk gait; this should be expanded to include more complex motions such as crawling or load-bearing in future. The simple reflexes used here can apply directly to a different gait (such as crawling) by simply calibrating the crawl to give Lillian a new baseline, and applying the same corrections used for the walk on a crawl. Load-bearing is even simpler, as it relates to a previous motion. Changes should not have to be made to the robot, but simply weights added to the robot, and the balancing reflexes should immediately begin to compensate for the extra weight.

Future work should address the initial assumption was that all joint movement would be coupled, investigating the possibilities and difficulties involved with allowing each foot or leg to be corrected separately. Differing threshold bounds were used in prior work [1], but have not been investigated here. They may improve the corrections made by the threshold algorithm, as it allows for corrections to be more or less sensitive without extra tuning. Finally, physical modifications to the robot could make it more difficult for the robot to balance, such as by adding weight at the head for extra sway.

This research provides an initial foundation for work looking into balancing reflexes with accelerometers, as it shows that it is possible to balance with only an accelerometer, and a simple control method. However, it also shows that these methods only work for reasonably simple balancing. More complex adjustments are not implemented well with these methods. Terrain such as the stepping field will require in addition more complicated (or at least more effective) means of balancing than either a PID or a threshold controller.

References

1. McGrath, S., Baltes, J., Anderson, J.: Active balancing using gyroscopes for a small humanoid robot. In Mukhopadhyay, S.C., Gupta, G.S., eds.: Second International Conference on Autonomous Robots and Agents (ICARA), Massey University (2004) 470–475
2. Kagami, S., Kanehiro, F., Tamiya, Y., Inaga, M., Inoue, H.: Autobalancer: An online dynamic balance compensation scheme for humanoid robots. In: Proceedings of the Fourth International Workshop on Algorithmic Foundations on Robotics. (2000)
3. Huang, Q., ming Zhang, W., Li, K.: Sensory reflex for biped humanoid walking. In: Proceedings of the 2004 International Conference on Intelligent Mechatronics and Automation. (2004)
4. Kulvanit, P., Srisuwan, B., Siramee, K., Boonprakob, A., Laowattana, D.: Team kmutt: Team description paper. In: Proceedings of the RoboCup 2006. (2006) Accessible at www.humanoidsoccer.org/qualification/KMUTT_TDP.pdf; accessed December 12, 2006.