ANNALS of Faculty Engineering Hunedoara – International Journal of Engineering
ISSN: 1584 - 2665

CONTROL SYSTEM CONCEPT OF THE SIX-LEGGED WALKING ROBOT

Tomáš MICHULEK[1], Aleš JANOTA[2], Juraj SPALEK[2]

[1] University of Žilina, Institute of Competitiveness and Innovations, SLOVAKIA
[2] University of Žilina, Faculty of Electrical Engineering,
Department of Control and Information Systems, Žilina, SLOVAKIA

ABSTRACT:

The paper deals with a concept of the control system implemented in the six-legged walking robot. The presented control structure is based on application of three hierarchically arranged control layers and the modular approach covering control modules controlling individual legs, a module synchronizing the legs movement and the highest level of the control system based on fuzzy logic. Because of a standard inter-module communication interface, any of the modules can be modified or replaced without seriously compromising the overall system performance Control algorithms applied to control the gait of the machine have been developed and tested using both virtual and physical prototype of the hexapod robot.

KEY WORDS:

Six-legged robot, control system, modular approach, multi-level hierarchy, fuzzy

1. INTRODUCTION

There are many mechanical structures that can be used for physical and/or virtual realization of robots. Some of more obvious of them that come in mind are arms, legs, and hands. The paper deals with control of a six-legged structural solution known as a hexapod. In general, more legs the robot has, less control effort it requires. Unlike less-legged robots that have to walk dynamically all the time the six legged robot can take advantage of statically stable movement.

There have been several possible approaches applied in history solving the control task of multi-legged robots. Classical control techniques were historically first techniques applied to control leg movements, e.g. the biped robot built by Kato around 1980. They worked with exact models, manageable provided that no substantial changes of parameters or important perturbations occurred. Later a hierarchical control was applied enabling on-line calculations and encompassing increasing control complexity resulting from a growing number of legs. A group of significant problems has also emerged in relation with different quality of terrains, existence of possible obstacles in them as well as other factors implying uncertainty and potentially making the system unstable [1]. Therefore much effort has been made to find and employ more robust control methods to solve these problems. As typical examples one could mention solutions based on neural networks or fuzzy logic [2]. Usage of the former approach in real time applications is usually difficult because of a lot of excessive calculations. The latter approach seems to be more advantageous thanks to less needed computation power. It has been also applied in the presented control concept. The paper does not address higher level algorithms, such as the path finding, the obstacle avoidance or the object manipulation.

2. THE CONTROL SYSTEM

The control system of the six legged robot must generally cope with the following tasks: stability control, gait generation, and sensors data analysis. To ensure the stability of the walking robot, the control system must contain a set of rules stating what can be done with the robot structure without the risk of compromising the stability. Gait generation is a crucial part of every walking robot control system. In order to make the robot move, trajectories for legs have to be generated and realized with the actuator system. To give the robot ability to cope with the environment, the control system has to analyze data from sensors. In this case the sensor system has consisted of six tactile sensors and one three axis accelerometer. The control system has got a modular design based on three basic layers:

control modules controlling individual legs, a module synchronizing legs and a higher level of the control system based on fuzzy logic. Because of a standard inter-module communication interface, any of the modules can be modified or replaced without seriously compromising the overall system performance. The overall concept of the control system of the walking robot is depicted in Fig. 1. It can be divided into the following three layers:

- Layer 0: Global fuzzy controller
- Layer 1: Central walk controller
- Layer 2: Individual leg controllers

The global fuzzy controller evaluates information from sensors and generates global movement strategy. The central walk controller serves as a connection between individual leg controllers, and is responsible for tasks that cannot be distributed among them. Each individual leg controller is responsible for the movement of a single leg. It computes algorithms that can be distributed among limbs of the robot.

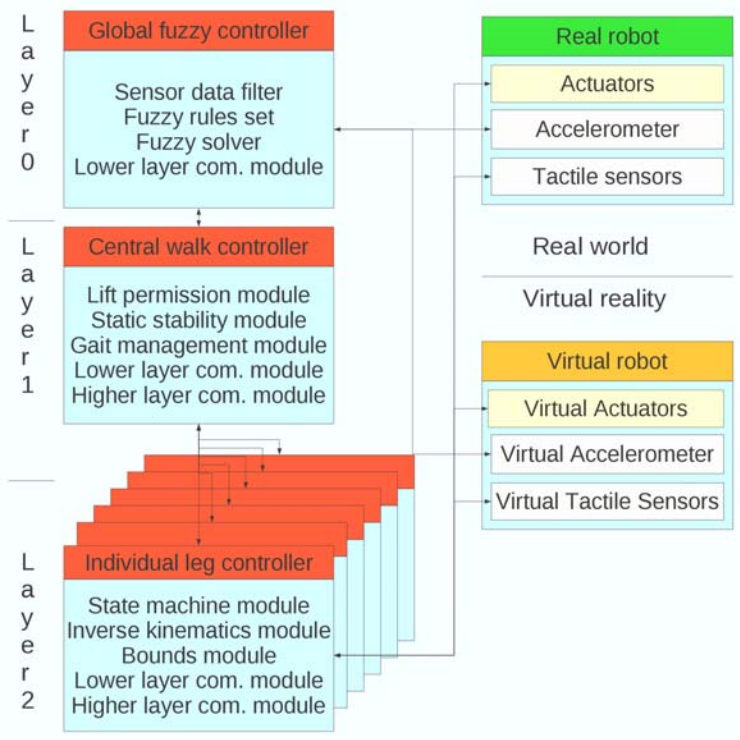

Figure 1. The control system structure

The central walk controller contains three basic modules: static stability module, gait management module and communication modules. The static stability control is based on a simple rule, that every leg can be lifted from the ground only with a granted permission from the central walk controller. This permission is given for leg lifts that won't compromise static stability of the robot. The control system predicts the stability margin S_m after the theoretical leg lift and if it fulfils the following condition, the central walk controller gives lift permission to the individual leg controller:

$$S_m > S_e \tag{1}$$

Where S_e is the minimum value the stability margin can reach, with the robot still maintaining the required amount of stability.

The described permission system also serves as an instrument for gait control. There are three predefined gaits the robot can perform: 1-5 gait, 2-4 gait and 3-3 gait. The 1-5 gait enables only one leg to be lifted at a time. This concept results in high stability combined with the low speed. This gait is supposed to be used in problematic situations such as very problematic terrain or obstacle. The 2-4 gait is a faster and less stable alternative to the 1-5 gait. It was designed for standard uneven terrain walking. The 3-3 gait is the fastest gait and its usage is limited to planar terrains with no possible stability issues. The basic rule for the 3-3 gait generation is that at least three supporting legs must be on the ground in order to make the lifting of any leg possible. The 2-4 gait requires at least 4 legs to be left on the ground after a leg lift.

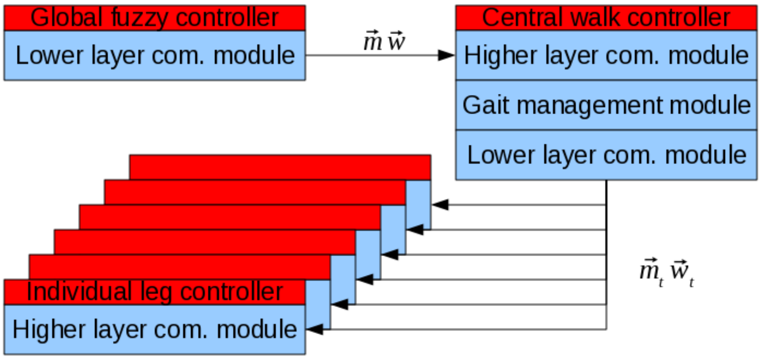

Figure 2. Movement and rotation transformations

The global fuzzy controller generates the movement requirements for the robot in the form of two vectors $\vec{m}$ and $\vec{w}$. Vector $\vec{m}$ represents expected movement direction and vector $\vec{w}$ holds the angular speed the robot should be rotating with. In the central controller these vectors are slightly modified accordingly to the current state of the drive train and the character of the terrain. Modified values $\vec{m}_t$ and $\vec{w}_t$ are then sent to individual leg control systems. The control algorithm has been designed to limit the speed and the angular speed of the robot accordingly to the terrain it walks on, to check if the expected values are not higher than maximum values the legged drive-train can deliver, and to send computed values of $\vec{m}_t$ and $\vec{w}_t$ to all individual leg controllers.

The individual leg controller generates position of the controller leg's sole. In order to prevent the controller from generating a sole position unreachable by the leg, a cylindrical shape is used to restrain the resulting sole position (Fig. 3). Possible states of the leg controller are A (on the ground), B (in the air), C (lifting) and D (laying). To perform a gait, each individual leg control system has to move periodically from one state to another, in the following order: A, B, C, D, A. The mechanism that induces state switching in the control system detects situations, when a leg sole is about to leave the 3-D structure and switch the state of the control system to prevent it. In the state C the leg is being lifted vertically and in state D it is being laid down on the ground. Movement of the legs must be limited to mirror the limits of the structure and actuators. The structure of the inverse kinematics solver is shown in Fig. 4. The individual leg controller output is in the form of the vector $\vec{l}_i$ representing the expected movement direction of the leg sole. In the inverse kinematics block, this vector is numerically integrated in order to get the expected position of the sole $\vec{s}_i$:

$$\vec{s}_i = \int_{t_0}^{t_c} \vec{l}_i \, dt \tag{2}$$

Where $\vec{s}_i$ is an expected position of the leg's sole in a coordinate system attached to the robot body, t_0 is the start time of the simulation and t_c is the current time in the simulation. The resulting sole position $\vec{s}_i$ is then transformed to expected angles of three leg actuators α_i, β_i, γ_i.

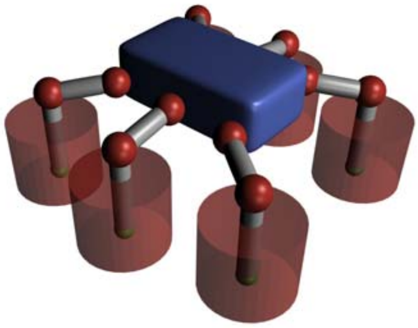

Figure 3. Restraining cylinders

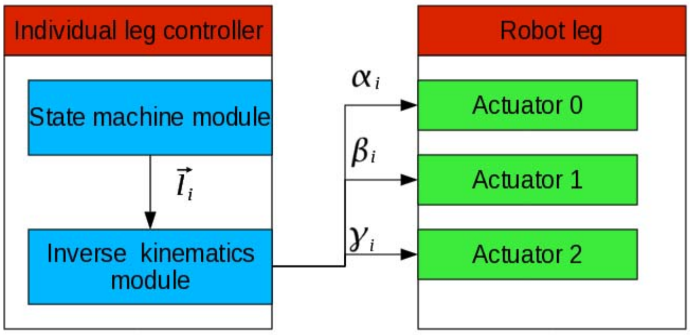

Figure 4. Inverse kinematics solver

The main task of the global fuzzy controller is to analyze information gathered by tactile and inertial sensors, and use results of the analysis to make assumptions about the character of the terrain the robot is walking over. These assumptions are used to pick the best gait type and movement speed for the particular terrain type. Two input variables are used in the fuzzy controller: body angle A_b and contact point difference D_v. Body angle A_b represents the angle among the horizontal plane connected to the robot body and a horizontal plane connected to the environment. In virtual reality this value is gained from virtual gyroscopic sensor, in the real world from a three axis accelerometer. The resulting angle is then fuzzified to set the value of a fuzzy variable. Contact point difference D_v is a variable that represents how rough the terrain is. The value is computed from data gained with tactile sensors:

$$D_v = |\ max(d_0 ... d_n) - min(d_0 ... d_n)\ | \tag{3}$$

Where n is the number of legs, and d_i is the vertical distance of the leg's sole from the centre of the robot body. This value is computed only for supporting legs. After the variables A_b and D_v were fuzzified a set of hand made rules is applied. As a result, values of the following two fuzzy variables are set: movement speed V_m and a gait type G. These variables are then defuzzified and results of the defuzzification are sent to the central controller.

3. CONCLUSIONS

Two versions of the robot have been created: a simulated virtual model and a real-world prototype having six legs, 18 degrees of freedom, 18 actuators, 3 tactile sensors per every leg, and 2 infrared sensors. Control algorithms used to control the gait of the machine were developed and tested virtually first [3], then the real-world tests on the prototype were performed. Future work would contain solving problems of diagnosis [4], optimization and further improvement of the concept.

Acknowledgements

The paper has been elaborated with support of the project "Development of modular mobile robotic systems (Vývoj modulárnych mobilných robotických systémov – VMROS), ITMS Code 26220220095.

REFERENCES

[1] *Celaya, E. – Porta, J.M.: Control of a six-legged robot walking on abrupt terrain. Proc. of the IEEE international conference Robotics and Automation, Vol. 3, 1996 (p. 2731-2736)*

[2] *Sanz, A.: A Six Legged Robot Controlled by ARS. University of Zaragoza, Department of Electronics and Communications Engineering, Spain. Retrieved from* http://www.cps.unizar.es/~te/Publicaciones_archivos/Ieefuz97.pdf

[3] *Michulek, T.: Using virtual reality to develop six legged walking robot control system. Applied Computer Science, Vol. 3, No. 2, 2007. (p. 52-62)*

[4] *Krokavec, D. – Filasová, A.: Dynamic Systems Diagnosis. Elfa, Košice, 2007*

ANNALS OF FACULTY ENGINEERING HUNEDOARA
– INTERNATIONAL JOURNAL OF ENGINEERING
copyright © University Politehnica Timisoara,
Faculty of Engineering Hunedoara,
5, Revolutiei, 331128, Hunedoara,
ROMANIA
http://annals.fih.upt.ro