ISSN 1392 - 1207. MECHANIKA. 2010. Nr.5(85)

Mechanical design and system control of quadruped robot

S. Vatau*, V. Ciupe, C. Moldovan***, I. Maniu******

**University Politehnica of Timisoara, Mihai Viteazu 1, 300222 Timisoara, Romania, E-mail: steliana.vatau@mec.upt.ro*
***University Politehnica of Timisoara, Mihai Viteazu 1, 300222 Timisoara, Romania, E-mail: vali.ciupe@mec.upt.ro*
****University Politehnica of Timisoara, Mihai Viteazu 1, 300222 Timisoara, Romania, E-mail: moldo80@yahoo.com*
*****University Politehnica of Timisoara, Mihai Viteazu 1, 300222 Timisoara, Romania, E-mail: manix@mec.upt.ro*

1. Introduction

Walking robots have potential advantages over traditional vehicles, and they have already succeeded in carrying out many tasks that wheeled or tracked robots cannot handle. Nevertheless, their use in industry and services is currently limited in scope [1].

Walking robots are, therefore, more suitable to perform tasks in environments that are sensitive to intrusion. A walking robot can also benefit from that active suspension is an intrinsic part of its structure and allows the robot to adapt to uneven terrain. This would allow for smoother ride for any passengers or cargo, and furthermore, if the robot were, for instance, equipped with a manipulator, the legs could provide an active but stable base while tasks are being performed. Furthermore, walking robots are omnidirectional robots, as this can walk forward, sideways, or turn on the spot, and additionally have the ability to raise or lower their body or tilt it, by varying the length of their legs, i.e. by bending their knees. This ability gives walking robots an advantage in manoeuvring through cluttered and tight environments. All the above-mentioned advantages of walking robots are dependent on the design of their mechanical structure and the control system [2]. There are many challenges with designing and building legged robots. The large number of actuated degrees of freedom makes them heavier, more complex and expensive than wheeled systems.

Most walking robots of today are quite slow and have bad payload weight-to-own-weight ratio compared to more conventional wheeled or tracked robots [3]. The control of a walking robot has to cope with a highly nonlinear system with many degrees of freedom, changes in the system dynamics as the legs are being lifted and placed, and unknown dynamics such as the interaction of the foot with the ground. Control of the walking robots also requires that the issue of stability against tipping over be treated in a more specific fashion than for wheeled robots, as there are discrete changes in the support of the robot when legs are lifted or placed [4].

One approach to designing legged robots is to study the solutions that nature has provided to the locomotion of animals through millions of years of evolution. The argument is that the way animals are built or move may in some sense represent an optimum, which may be beneficial in the design and control of legged robots. For instance the study of gaits has led to mathematical models that can explain why some gaits are preferred by animals. Furthermore, the development of very simplified mechanisms, in terms of number of degrees of freedom and actuators, has also proven that legged locomotion can be accomplished without trying to mimic the complexity of animals [1].

2. Robot design and locomotion module

The robot presented in this paper, named JQuadRobot has 12 degrees of freedom (DOF), with three degrees of freedom per leg. Each leg has hip, knee and ankle. The hip joint is actuated in vertical plane (Pitch) and horizontal plane (Roll), knee joint is actuated in vertical plane (Pitch) and ankle is not actuated. Fig. 1 shows the quadruped robot model and Fig. 2 shows the real robot. The mechanical design is divided into four phases: determining the mechanical constraints, conceptual design, building the prototype model, specification and fabrication of the model. There are various design considerations when designing a quadruped robot. Among them, the major factors that have to be considered are robot's size selection, DOF selection, link design, stability and foot pad design. Size robot plays a major role. Based on this the cost of the project, materials required for fabrication and the number of actuators required can be determined. In this project miniature size of the robot is preferred so a height of 500mm is decided which includes mounting of the control circuits, but the actual size of the robot is 400mm without controlling circuits. The leg has got six degrees of freedom (Hip–3 DOF, Knee–1 DOF, Ankle–2 DOF), but implementing all the six DOF is difficult due to increase in cost of the project and controlling of the actuators which become complex, so in this project reduced degrees of freedom are aimed so 3 DOF per leg has been finalized.

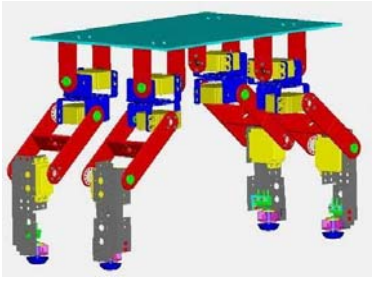


Fig. 1 CAD model of the robot



Fig. 2 The real model of the robot

With quadruped mechanism, three points will be in contact with the ground surface. The stability of the robot is determined by the foot pad. In a statically stable gait, the vertical projection of the centre of gravity (G) onto a horizontal plane is kept within the support area at all times, as shown in Fig. 3. In the absence of any inertial or external forces and if the ground is sufficiently rigid, the robot can remain stable as long as the G is within the support area [5]. For robots with point feet, a necessary condition for static stability is that the robot has at least three legs on the ground at all times. This is necessary in order to form an area of support that can contain the projection of G within its borders.

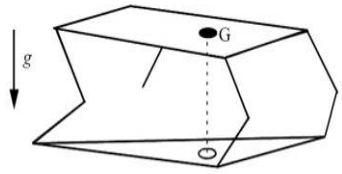


Fig. 3 Vertical projections of feet contact points and centre of gravity (G) on a horizontal plane

In Fig. 4, in the left part of the figure, three legs provide support and the projection of the centre of gravity is located inside the support area such that the robot is statically stable. The foot placement in the right part projects the centre of gravity outside the support area, which leads to instability due to a tipping moment caused by gravity.

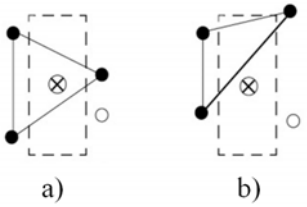


Fig. 4 Support polygon: a) statically stable case; b) statically unstable case

Generally there is a concept that over sized and heavy foot pad will have more stability due to more contact area. But there is a disadvantage in using the oversized and heavy foot pad, because the torque requirement of the motor is more and lifting the leg against the gravity becomes difficult. By considering this disadvantage an optimal sized foot pad was used.

Stable walking pattern can be obtained only if the centre of mass and the centre of pressure are within the supporting [2]. Generally walking cycle consists of two steps namely “initialization” and “walking”. In the “initialization” step the robot will be in balanced condition and in this step the servomotors are made to return to home position. This will certainly help the robot to advance into the next step.

3. Kinematics and trajectory planning

The quadruped cannot make another step, without bringing the support leg in the support polygon corners. The programmed routines elaborated to implement the configuration operations into the control software are based on the leg's inverse kinematic model (Fig. 5).

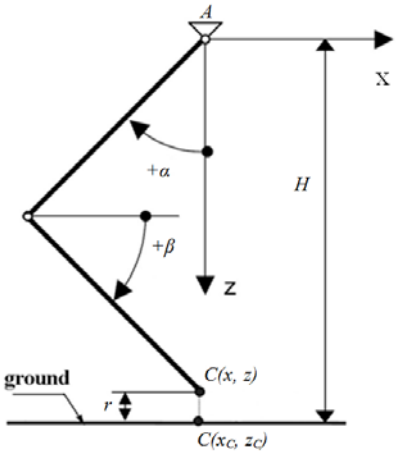


Fig. 5 Inverse kinematic model

The inverse kinematic model supposes that the support polygon's corner points $C(x_C, z_C)$ are defined. The unknown variables are the orientation angles α and β. According to the algorithm [6] the angles are determined with the relations (1)

$$\left.\begin{aligned} \alpha &= arcsin\,\frac{u}{l_{AB}} \\ \beta &= arcsin\,\frac{v}{l_{BC}} \end{aligned}\right\} \qquad (1)$$

where

$$v = \frac{b}{a} - \sqrt{\frac{b^2}{a^2} - \frac{c}{a}}$$

$$u = -\frac{x}{z}v + \frac{x^2 - z^2 + l_{BC}^2 - l_{AB}^2}{2z}$$

$$a = \frac{1 + x^2}{z^2}$$

$$b = \left[1 + \frac{\left(x^2 - z^2 + l_{BC}^2 - l_{AB}^2\right)}{2z^2}\right] \cdot x$$

$$c = \left[\frac{\left(x^2 - z^2 + l_{BC}^2 - l_{AB}^2\right)}{2z}\right]^2 + x^2 - l_{AB}^2$$

$$x = x_P$$

$$z = z_P - r = H - r$$

During the configuration, the leg’s characteristic point goes through a fragmented trajectory (Fig. 6). This rajectory has the C_1 as the start point and the C_2 as the stop point, in successive positions of the ankles joint, before

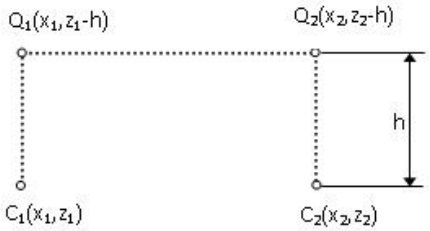


Fig. 6 A fragmented trajectory

and after the configuration operation. The fragmented trajectory contains the first rising segment C_1Q_1, the second horizontal movement segment Q_1Q_2, and the third descending segment Q_2C_2.

This algorithm associates to each coordinate pair, one angle pair. With the four pair of angles (α_{C1}, β_{C1}), (α_{C2}, β_{C2}), (α_{Q1}, β_{Q1}), (α_{Q2}, β_{Q2}) the angular range, needed in the hip joint and the knee joint is defined by the relations (2)

$$\left.\begin{aligned}
\Delta\alpha_{C1Q1} &= \alpha_{Q1} - \alpha_{C1} \\
\Delta\alpha_{Q1Q2} &= \alpha_{Q2} - \alpha_{Q1} \\
\Delta\alpha_{Q2C2} &= \alpha_{C2} - \alpha_{Q2} \\
\Delta\beta_{C1Q1} &= \beta_{C1} - \beta_{Q1} \\
\Delta\beta_{Q1Q2} &= \beta_{Q2} - \beta_{Q1} \\
\Delta\beta_{Q2C2} &= \beta_{C2} - \beta_{Q2}
\end{aligned}\right\} \quad (2)$$

Every configuration routines, used by the control software to change the walking method, are based on the same relations like (2).

In the walking locomotion, the system is permanently reconfigured. This process depends on both the stepping leg's movement and on support leg's movement. In order to formulate the algorithm, that permits to include these influences into the control routine, the trajectory of the characteristic point is broken in respect of the frame, in two branches. In the first branch the stepping phase related part is kept, and in the second the support phase related part. This operation is called brute segmentation and is presented in Fig.7, a. The branch passed in the stepping phase, contains the same segments C_1Q_1, Q_1Q_2, Q_2C_2 like the configuration trajectory that was discussed for the immobile supports. Therefore the afferent movements for this branch can be controlled also by routines based on the described algorithm. The branch passed during the support phase (C_2C_1 segment), is passed in three sequences, which on duration rule has the same lengths equal to $L/4$. In the support phase of the leg's 2, the first sequence corresponds to the C_2C' interval - when the leg 4 is stepping, the second sequence corresponds to the interval $C'C''$ - when the leg 1 is stepping, and the third sequence corresponds to the interval $C''C_1$ - when the leg 3 is stepping. After the brute segmentation a fine segmentation is made (see Fig.7, b). This operation divides each support segment in subintervals, i.e. $C'CQ_1$, CQ_1CQ_2, CQ_2C'' having the lengths proportional with the other segments C_1Q_1, Q_1Q_2, Q_2C_2 from the stepping phase branch (because they have constant speeds).

According to the two segmentations, for the subinterval lengths, the relations (3) are obtained. These relations permit to calculate the coordination of the intermediary points CQ_1, CQ_2, and with the inverse geometrical algorithm to deduce the joints ranges, needed to correlate each support leg movement with the other stepping leg movement

$$\left.\begin{aligned}
l_{C'CQ1} &= l_{Q2C''} = \frac{hL}{8h+3L} \\
l_{CQ1CQ2} &= \frac{L}{4} - 2\frac{hL}{8h+3L}
\end{aligned}\right\} \quad (3)$$

These calculations must be repeated for each support interval and for each involved leg, in both the support phase and in the stepping phase. They use the same algorithm and software routine, only with changed dates (the coordinates of the trajectory points).

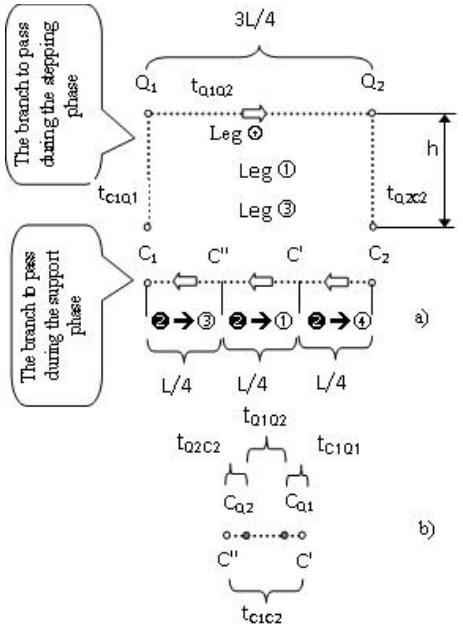


Fig. 7 Trajectory: a) brute segmentation b) fine segmentation

3. Robotics' control system

For monitoring and controlling a custom mobile robot named JQuadRobot, the author has developed a Graphical User Interfaces (GUI) (Fig. 8) [7].

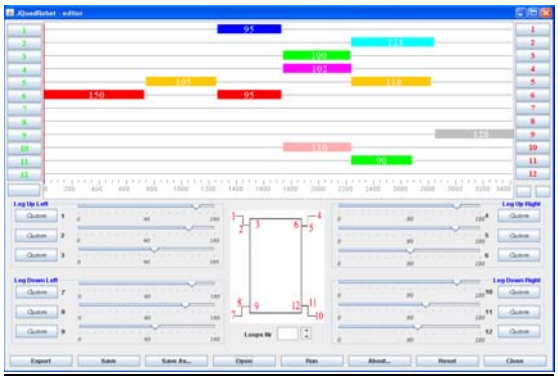


Fig. 8 GUI for monitoring and controlling a JQuadRobot

This GUI is developed in Java and is essential in development motions for the legged robot. The friendly interface, allows any user to define and test movements for this robot. The cross-platform capability was the first reason to choose Java language for developing this application. The application is released as free software under the GNU General Public License (GPL) and was conceived modular (Fig. 9).

This GUI allows controlling the robot by two methods. Using the first method, in the top part of the GUI, the user can add (by selecting one of the button of the left part) or remove (by selecting one of the button of the right part) one of the motors corresponding to the robot. Adding a motor allows also setting the rotation (in degree) and how long time (in ms) that rotation must be performed.

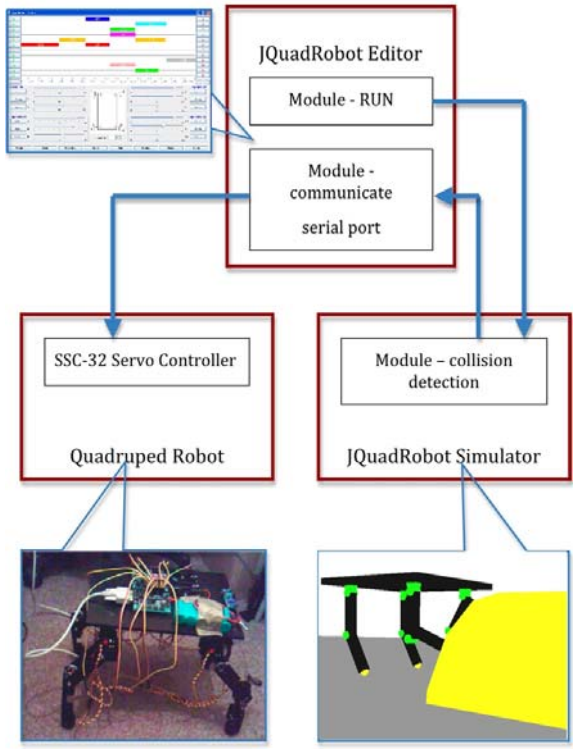


Fig. 9 Modular application JQuadRobot

The motor's time of operation is show by a colour rectangle (see Fig. 10). In this way it can be possible to define a gait diagram (i.e. pace, walk, trot, or gallop).



Fig. 10 The top part of GUI for control a JQuadRobot

After the user defines a gait diagram, it is possible to save all information in a file with XML extension using a "Save As" button (Fig. 11). In this mod, the user can define the forward, backward, turn left and turn right walking, concordant a different gait.

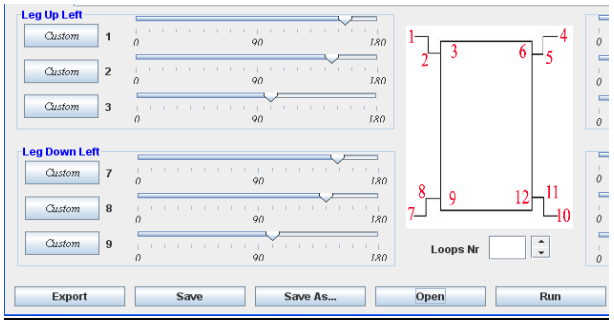


Fig. 11 Left bottom part of GUI

The "Open" button is in the bottom part also, and permits the user to open an existing file. The GUI has a "Run" function, which gives the user a possibility to move the robot in concordance with its settings. With a "Reset" function, the user set the default value. Another method to move the legged robot is by using the slider for rotation and two slider bars to control the time (start and stop) corresponding each motor marked with a number from 1 to 12.

The user can set the sense of motors rotation (CCW or CW) and how many steps execute the robot using "Loops Nr" (Fig 12). The "Custom" button allows customize the angular velocity, represented by a graph. Thus, using the last method any adjustment is sent in real time to the robot, which performs the commands.

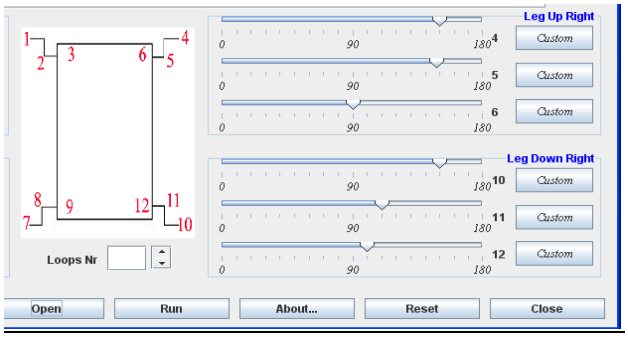


Fig. 12 Right bottom part of GUI

By the second method, moving of the legged robot is performed using the sliders for rotation corresponding each motor marked with a number from 1 to 12. In Fig. 9 the sliders and the positions of servomotors represented on schema on left part can be seen.

The user can combine different kind of movement and can export the configuration in an xml file or in format for microcontroller (Fig. 13).

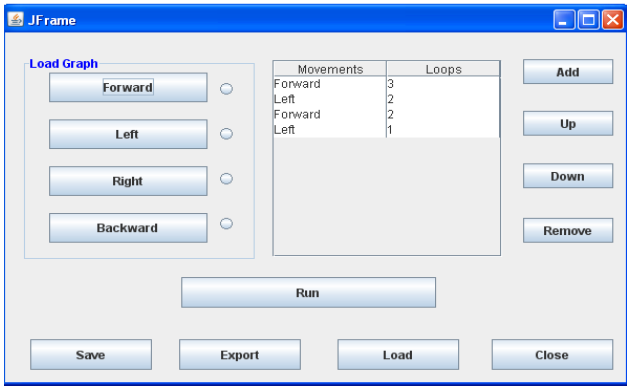


Fig. 13 Export windows of GUI

The simulator can be used together with JQuadRobot Editor to prevent collision detection between robot's legs by running the commands before sending it to robot. Another feature is to make constrictions to prevent collision detection in some environments like walking through a pipe or through a frame with specific dimensions.

In the Fig. 14 the simulator searching a way to pass an obstacle is presented. The algorithm for crossing the obstacle follows the next rule: the robot lifts its leg, and beats the obstacle. If there is an obstacle, the robot increases up leg and verifies again. When there are no more obstacles in face, then the robot moves in concordance with the strategy to the displacement over obstacle.

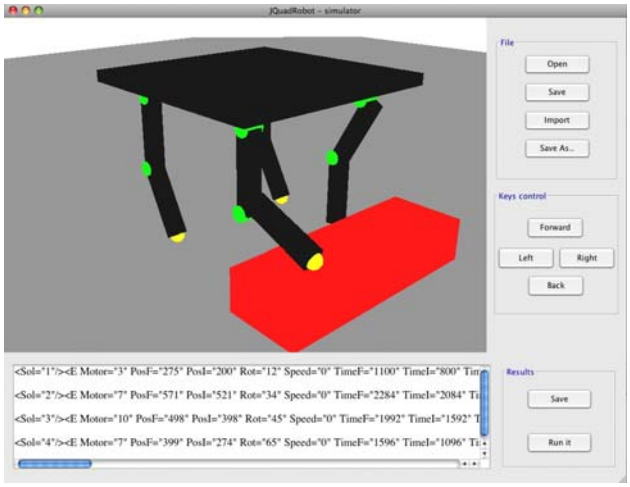


Fig. 14 Simulator application JQuadRobot

5. Conclusion

Quadruped robots are the fundamental block of any advanced walking robots. By making the quadruped robots fully autonomous, these can be used in environment where human cannot enter. Based on the analysis and study, the output of this type of robots can be used for developing artificial limbs for the physically challenged person.

6. Future work

The future advancement can be carried out in the project by going for embedded processor that can process and transmit the control signal faster to the actuators. Complex movements can be achieved by increasing the DOF. Vision system can help the robot to work autonomously. Remote control through wireless ethernet mode can also be considered.

References

1. **González-de-Santos P., Garcia E., Estremera J.** Quadrupedal Locomotion: An Introduction to the Control of Four-legged Robots. -Publisher Springer, 2006, ISBN-10: 1846283078, ISBN-13: 978-1846283079.
2. **Buehler, M., Battaglia, R., Cocosco, A., Hawker, G., Sarkis, J., Yamazaki, K.** Scout: A simple quadruped that walks, climbs and runs. -Proceedings of ICRA98, 1998, p.1707-1712.
3. **Nitulescu M., Stoian V**. Modelling and control aspects of specific mobile robot. -Mechanika. -Kaunas: Technologija, 2008, Nr.1(69), p.54-58, ISSN 1392 - 1207.
4. **Ilg, W., Albiez, J., Jedele, H., Berns, K., Dillmann, R.** Adaptive periodic movement control for the four legged walking machine BISAM", 1999, Available from http://ieeexplore.ieee.org Accessed: 2008-03-10.
5. **Fukuoka, Y.; Kimura, H., Cohen, A. H.:** Adaptive Dynamic walking of a quadruped robot on irregular terrain based on biological concepts. -The International Journal of Robotics Research, Vol. 22, No. 2, February 2003, p.187-202.
6. **Vatau, S., Varga, S., Radulescu, C**. Algorithms for the quadruped mobile robot locomotion system configuration.-Proceedings of International Conference Optimization of the Robots and Manipulators, p.161-167, ISBN 978-973-648-656-2, Predeal, Romania, May 2007, Bren Publishing House, Bucureşti.
7. **Vatau, S.** The constructive - functional optimization for a quadruped mobile robots" (Romanian language), PhD thesis, Ed. Politehnica, Timisoara, 2008, ISSN: 1842-7707, ISBN: 978-973-625-613-4.

S. Vatau, V. Ciupe, C. Moldovan, I. Maniu

KETURKOJO ROBOTO MECHANINIS PROJEKTAVIMAS IR VALDYMAS

R e z i u m ė

Mobilios platformos naudojamos programuotam arba nuotoliniam judesio valdymui daugelyje sričių. Erdviniam judesiui naudojamos dvi sistemos: riedėjimo (ratai arba bėgiai) ir vaikščiojimo. Šiame darbe pristatomas keturkojis robotas „JquardRobot". Šio projekto tikslas – sukurti platformą dinaminiam vaikščiojimui ir dirbtinam intelektui tirti. Straipsnyje pateiktas mechaninis projektas ir keletas valdymo sistemų kontūrų.

S. Vatau, V. Ciupe, C.E. Moldovan, I. Maniu

MECHANICAL DESIGN AND SYSTEM CONTROL OF QUADRUPED ROBOT

S u m m a r y

The mobile platforms are used to perform programmed or remote controlled movement in a very large working space. In the terrestrial locomotion two systems are used: based on rolling (wheeled and tracked), or based on walking. The research work on a small size quadruped robot: JQuadRobot is presents. The objective of this project is to build a platform for the study of dynamic walking and artificial intelligence. In this paper, the mechanical design and some control system configuration aspects are presented.

С. Ватау, В. Цюпе, С. Молдован, И. Маниу

МЕХАНИЧЕСКОЕ ПРОЕКТИРОВАНИЕ И УПРАВЛЕНИЕ ЧЕТЫРЕХНОГОГО РОБОТА

Р е з ю м е

Мобильные платформы используются для осуществления программного или дистанционного управления во многих отраслях. Для пространственного движения используются две системы: качения (колеса или рельсы) и шагающая. В данной работе представлен четырехногий робот JquardRobot. Цель этого проекта создать платформу для изучения динамического шагания и искусственного интеллекта. В настоящей работе представлен технический проект и несколько контуров управления.

Received May 27, 2010
Accepted October 11, 2010