

Development of a Self-Stabilizing Robotic Chassis for Industry

Igor Ryadchikov¹, Semyon Sechenev^{1,*}, Alexander Svidlov¹, Sergey Sinita¹ and Evgeny Nikulchev^{2,3}

¹Kuban State University, 350040 Krasnodar, Russia

²Moscow Technological Institute, 119334 Moscow, Russia

³Moscow Technological Institute MIREA, 119454 Moscow, Russia

Abstract. Presented the description of the bipedal robotic chassis with the unique kinematic scheme which has the possibility to locomote in complicated multi-level environment. AnyWalker is equipped with the system of compensation of external impacts with motor-wheels which can self-stabilize the robotic system in 3 dimensions. Presented chassis suggests to have open software and hardware architecture in order to become the universal walking platform for service and industry robots.

1 Introduction

Service and personal robotics sets huge amount of tasks which require the ability of locomotion in complicated environment, sometimes multi-level with stairways, doors, windows, city obstacles. Sometimes there is no way to specially prepare the field of action i.e. make the surface flat or remove the obstacles. Therefore if the cost of environment preparing is too high there is the idea of creating the universal robotic system that is able to move in various surroundings.

The problems of creating anthropomorphic robots with the same purpose are discussed in [1]. There are no analytically solvable mathematical models of bipedal robots without independent stabilization systems. The task of creating the mechanical solution of anthropomorphic robot with the same degrees of freedom as human's is not solved yet.

It is proposed to encapsulate the bipedal locomotion function in complicates environment the robotic chassis with the possibility of hardware and software extension due to open architecture standarts.

2 Kinematic schemes

Bipodal robotic chassis is proposed, which has the ability to locomote in complicated not specially prepared environment and has the independent encapsulated built-in stabilization module.

The advantages of the kinematic scheme shown in Fig. 1:

- Large fiend of static stability zones;
- Low center of mass comparing to anthropomorphic design;
- The ability to roll on body and balance using segway principle and manipulate objects with two "hands";

- The ability to climb on the obstacle three times higher than the robot itself in a default position;
- The ability to stabilize standing on one "leg" and manipulating objects with the other;
- Better density comparing to the anthropomorphic construction having similar step size.

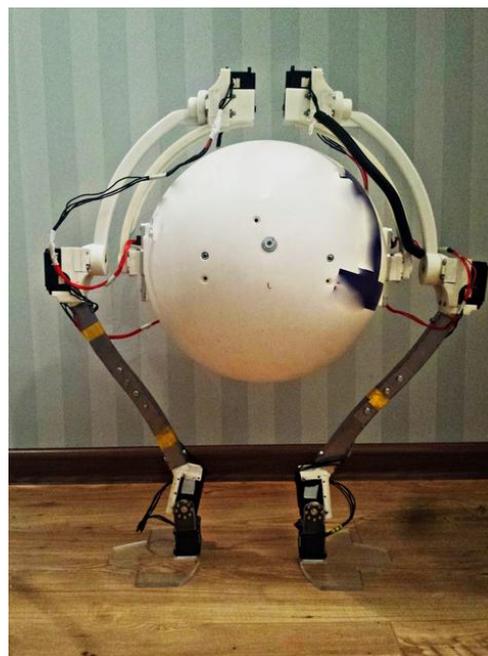


Fig. 1. General view of the prototype

Kinematic scheme is shown on the Fig. 2, where 1 – body of the robot, 2 – control unit, 4 – power supply, 5 – corrective center of mass displacement unit, 6 – surface, 7 – axis of the first and second hinges rotation, 8 – axis of gravity passing through the center of mass, 9 – obstacle, 11 – first part of the upper robot leg, 12 – second part of the upper robot leg, 13 – lower robot leg,

* Corresponding author: sechenev.semen@gmail.com

14 – foot, 21 – first hinge, 22 – second hinge, 23 – third hinge, 24 – fourth hinge.

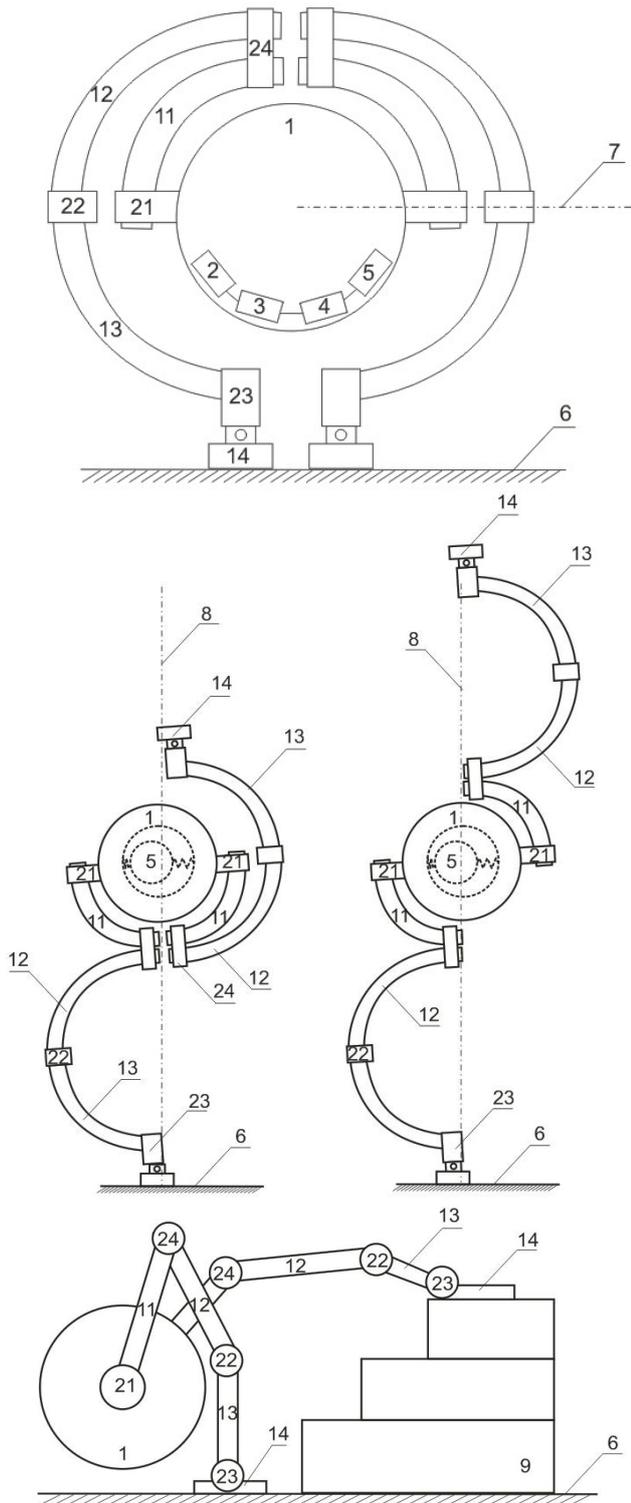


Fig. 2. Kinematic scheme

AnyWalker can reach continuous speed of 3 km/h and peak speed of 7,7 km/h with the form-factor of 0.3 m ball diameter while walking and 1.5 km/h while rolling.

With the board voltage of 16 V, servos average current assumption of 2.5 A, average speed of 0.8 m/s,

stabilization system power of 2 kW and robot weight of 10 kg there is the cost of transportation equals to 1.85, where $COT = \text{energy} / (\text{weight} * \text{distance})$, which includes energy to run the motors and all electronics.

Comparisons:

- Toyota Prius COT = about 0.15
- Human COT = about 0.2 (a bit better than Ranger)
- Asimo COT = about 2 (54 kg, 1.5 m/s, 1.8 kW)
- BigDog COT = about 15 (12.8 miles, 4 gal gas, 110 kg)

3 Self-stabilization system

AnyWalker unique self-stabilization system consists of three motor-wheels, incapsulated built-in independent module, situated in orthogonal planes, and the centers of mass flywheels are the same. This stabilization system is inspired by the construction presented in [2-4].

While designing the AnyWalker chassis the problem of optimal parameters of the system selection is being faced. In particular, the task of deciding which flywheels to use in the design of the stabilization system, with the target optimization function of minimizing the weight of service structures, while maintaining the size of the field of management.

The problem of picking up a lighter flywheel geometrically similar to original which will improve the quality of one-dimensional stabilization system with the parameters proposed in [4] has the following solution. The original flywheel has mass $m_w = 0.204 \text{ kg}$ and inertia moment $I_w = 0.57 \times 10^{-3} \text{ kg}\cdot\text{m}^2$. Let k be the linear scaling factor of the flywheel, then the flywheel mass $m_w(k) = 0.204 k^3 \text{ kg}$ and inertia moment $I_w = 0.57 \times 10^{-3} k^5 \text{ kg}\cdot\text{m}^2$. The maximum deviation angle α_{\max} which the stabilization system can compensate dependence of k plot was built with the numeric methods described in [5]. It is seen that the $\alpha_{\max}(k)$ function reaches maximum value at $k \approx 0.6$. The conclusion is that for the stabilization system with such parameters there is the optimal flywheel with mass $m_w = 0.044 \text{ kg}$ and inertia moment $I_w = 0.0443 \times 10^{-3} \text{ kg}\cdot\text{m}^2$.

The following example (Fig. 3) shows the dependencies of the maximum deviation angle α_{\max} of k with the different friction coefficients.

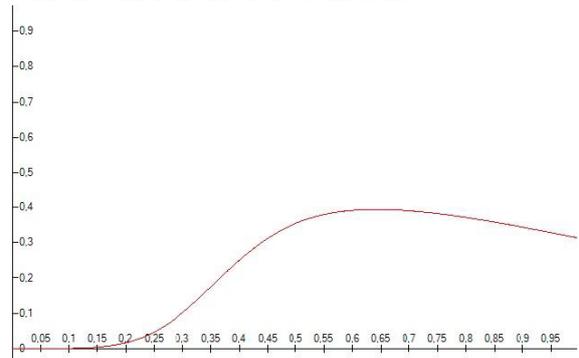


Fig. 3. The maximum deviation angle α_{\max} which the stabilization system can compensate dependence of k plot

Fig 4 shows three modelling experiments, in which all of the parameters except the friction coefficients are the same. The green plot accords to the experiment friction coefficient C , blue – $10 \times C$, red – $50 \times C$. The explanation of the plots is that the higher friction coefficient gives more restrictions to changing the parameters of the system with keeping the stabilization of the analytical solution of differential equations system.

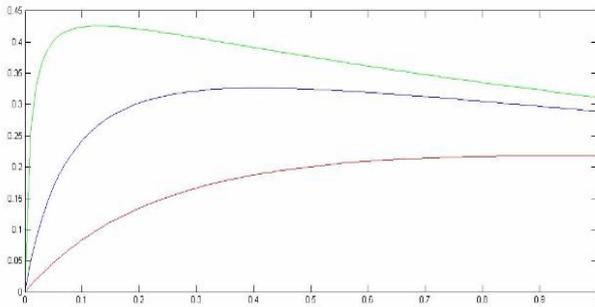


Fig. 4. Modelling experiments

Fig 5 shows the MATLAB simulation of the 2-axis stabilization with one «foot» hard connected to the ground.

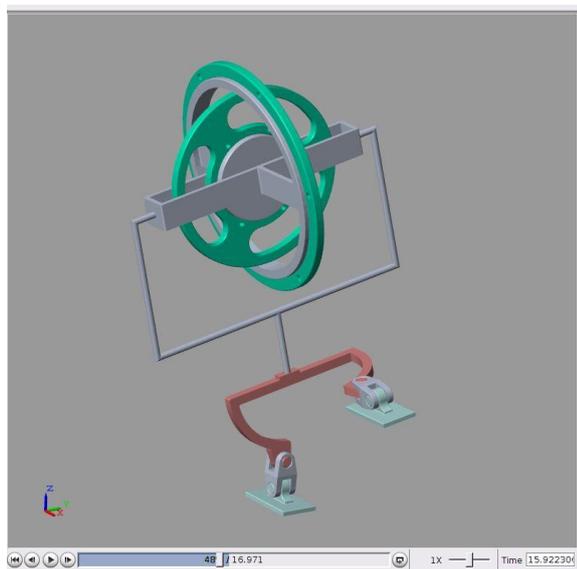


Fig. 5. Simulation of the 2-axis stabilization system in MATLAB

4 Hardware and software architecture of the control system

Hardware and software architecture of the control system AnyWalker consists of the following levels:

1. Reactive layer is based on a microcontroller STM32F429ZI. The lower reactive level performs robot stabilization using flywheels and robot movements using leg servos in a decoupled way. It receives the movement commands for the servos from Execution level and tries to run them keeping robot balance via flywheels,

monitors servos temperature and power consumption. This level is implemented on the STM32F429 Discovery microcontroller. IMU6050 based accelerometers/gyroscopes are queried via I2C bus and filtered via Madgwick sensor fusion algorithm. Flywheels movement and speed computation are implemented using Maxon EPOS2 controllers with Maxon XXX motors. Dynamixel XXX servos for robot legs are controlled via RS485 using MAX485 based converter.

2. The Executive level is based on the controller TRIK. The Execution level is connected to Reactive level via UART or Bluetooth. We created the mobile application for Android platform which sends movement commands to AnyWalker and displays the result. TRIK and ROS variants are planned.

3. Algorithmic level represents the application software installed on a personal computer or smartphone, completely or partially in the cloud. The Application level is planned to provide high level user interface, implement cloud services client API to recognize voice commands, gather sensor and commands data for machine learning purposes, connect several AnyWalkers with patterns of crowd behaviour.

It is proposed to use the control algorithmic support as an information service that will allow you to give third-party API researchers to solve applied problems.

The levels of the software architecture provide an API to the next level according to their functionality [5]:

1. Reactive level: the RPC API;
2. Executive level: C++ API, API QtScript;
3. Algorithmic level: the SOAP API.

AnyWalker chassis is presented to be a chassis for applied robotics with the following expansion possibilities:

Hardware level: adding different sensors, changing manipulators, increasing or decreasing the robot size. Provides the ability for third-party engineers to develop and sell their own hardware that is compatible with AnyWalker;

Control level: programming new movements, specialization, programming new manipulators, adding degrees of freedom.

Intelligent systems level: designing crowd AnyWalker patterns, organizing robotic competitions, artificial intellect systems on the basis of the chassis. Provides the ability for developers to create and implement their own applications.

5 Conclusions

Proposed bipodal robotic chassis is able to locomote in complicated not specially prepared environment and has the independent encapsulated built-in stabilization module and non-anthropomorphic design.

Proposed kinematic scheme allows to have cost of transportation around 2 with the diameter of robot body of 0.3 m and robot weight of around 10 kg, which is similar to the CoT of the Asimo robot.

Proposed optimization methods and criteria of the designed systems with stabilization based on the principles of the inverted pendulum with flywheel. Proposed stabilization system parameters calculation method allows minimizing the weight of service structures thus increasing the robotic system payload. Proposed scheme of open software and hardware architecture of the platform provides the possibility to create extensions and build various applied robotic systems based on AnyWalker locomotion module.

References

1. N.A. Chaturvedi, A.M. Bloch, N.H. McClamroch. *Proceedings of American Control Conference*, 3612-3617 (2006)
2. M. Muehlebach, G. Mohanarajah, R. D'Andrea, *2013 IEEE 52nd Annual Conference on Decision and Control (CDC)*, 1283-1288 (2013)
3. M. Gajamohan, M. Muehlebach, T. Widmer, R. D'Andrea. *2013 European Control Conference (ECC)*, 268-274 (2013)
4. M. Gajamohan, M. Merz, I. Thommen, and R. D'Andrea, *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3722-3727 (2012)
5. I. Riadchykov, S. Sechenev, S. Sinitsa, E. Nikulchev, *6th Seminar on Industrial Control Systems: Analysis, Modeling and Computation*, **6**, 01003 (2016)