Realization, Programming and Controlling of the Stewart-Gough Platform

Submitted: 9th July 2019; accepted: 7th January 2020

Dawid Owoc, Krzysztof Ludwiczak, Robert Piotrowski

DOI: 10.14313/JAMRIS/2-2020/14

Abstract:
This paper presents realization, programming, and controlling of a low cost Stewart-Gough platform (SGP) with rotary actuators. The realized SGP is applied in a ball & plate control system. Developed dedicated software consists of embedded and application software for both the SGP positioning system and the ball & plate control system. A ball position is being obtained using computer vision. The paper contains tests results for both an SGP positioning accuracy and a control quality of the ball & plate control system.

Keywords: *Stewart-Gough platform, ball & plate, computer vision*

1. Introduction

This paper presents a real implementation of the design of the Stewart-Gough platform (SGP) [3], [8] presented in the paper [6]. It describes realization, programming and controlling of the SGP.

The realized SGP is applied in a ball & plate control system. A ball & plate system is an expanded version (two degrees of freedom) of a ball and a beam system.

In this paper, a ball position is being obtained using computer vision. Utilizing an SGP in a ball & plate control system is common. For example, the paper [1] describes building of an educational kit of this type.

The paper is organized as follows. In Section 2, a realization process with photos of the assembled system is presented. In Section 3, developed software is illustrated. In Section 4, test results are provided. Section 5 contains project summary and conclusions.

2. Realization of the Platform

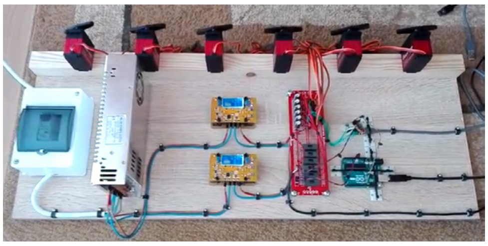

Fig. 1. A wooden board with all of the electronics and six servo motors for testing

Having designed the electronic subsystem in EAGLE environment [7], all the electronic parts were fixed to a wooden board. In the begining, a 10 A AC circuit breaker, an AC/DC power supply, and two step-down voltage converters were fixed. At the early stage of the realization, six servo motors (PowerHD HD-1235MG) were fixed to the board for testing. The wooden board is presented in fig. 1. After testing, the servo motors were moved to their target place - the base. Next, a perfboard with soldered electronic circuit was fixed. Every single servomotor was safeguarded with a dedicated fuse. Last two elements that were fixed to the board were: servo controller (Pololu Micro Maestro) and Arduino UNO.

Fig. 2. A fully assembled SGP

Fig. 3. A complete ball & plate control system research station

Having designed the mechanical subsystem in Inventor environment [5] all mechanical parts were aggregated. Metal elements such as servo motors mounting brackets and horns were made using laser cutting. The base and the moving platform with the plate for the ball were made with wood. Having all of the parts, the main assembling were started. In the begining, the base was fixed to a bigger rectangular plate. Next, horns were fixed to the servomotors. The servomotors were fixed with mounting brackets that ensure them not to move even at high load.

Assembling required high precision because every error during assembling woud have a big impact on the SGP's geometry and as a result on a correctnes of the inverse kinematics problem (IKP) solution [6].

Having assembled the base, the assembly of the moving platform and connecting it to the rotary actuators with six legs was started. An important attribute is the range of angle of the ball joints. If the maximum angle was too low it would reduce the workspace of the SGP. The ball joints used in the assembled SGP has a maximum angle of 40° which is sufficient.

The last part of the realization process involved creating the ball & plate control system research station. A camera was hung on a tripod. All of the devices were connected to the Personal Computer (PC). Fig. 2 presents a fully assembled SGP and fig. 3 presents a complete research station.

3. Programming of the Platform

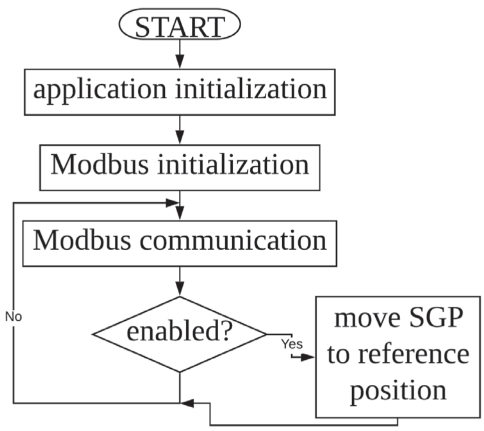

Fig. 4. A flowchart of the embedded software algorithm

All software was developed in C++. The software consists of embedded software and PC software. The embedded software runs on Arduino UNO with an AVR microcontroller. The PC software runs on Debian operating system (OS).

In order to facilitate the process of building the application, CMake environment [4] was used. CMake is an open-source, cross platform build system.

The embedded system is the lowest abstraction layer of the system. It is the only one that has a direct access to rotary actuators. It calculates the IKP [6] and provides safety functions to prevent danger conditions, e.g. trying to force a position and/or orientation beyond the workspace. A flowchart of the algorithm of the embedded software is presented in fig. 4.

The PC software allows a user to position the SGP using command line interface (CLI) and high abstract

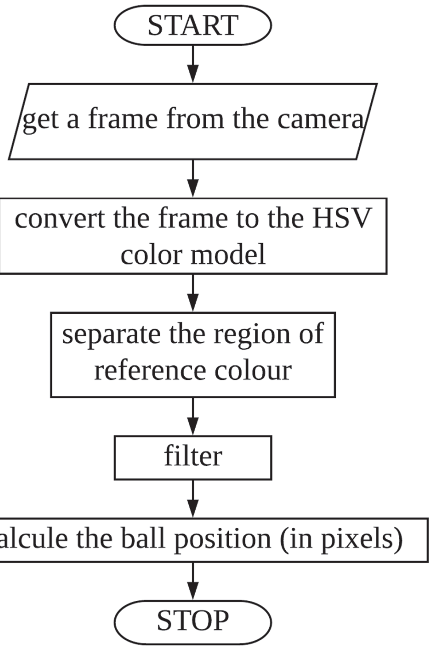

Fig. 5. A flowchart of the ball vision application algorithm (one iteration)

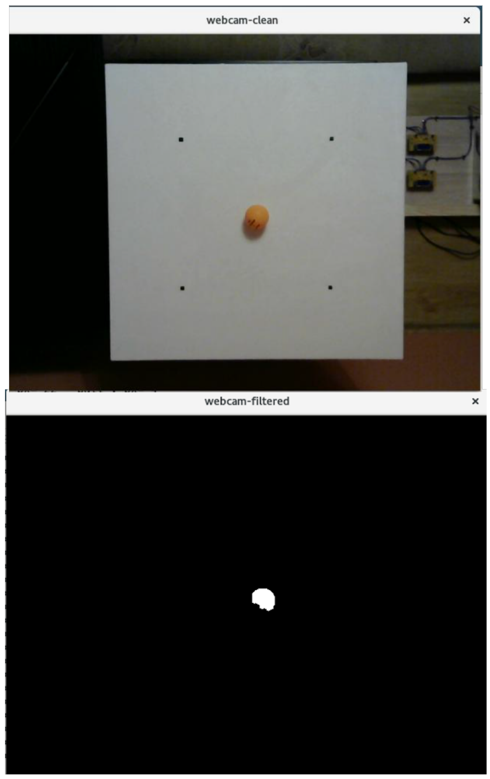

Fig. 6. Screenshots presenting view from the camera - raw and with ball detected

commands, e.g.: *–set-platform –pitch 10 –roll 25* to rotate the SGP to $\mathbf{R} = \begin{bmatrix}0 & 10^\circ & 25^\circ\end{bmatrix}$. External applications can benefit from this software using a library with relevant Application Programming Interface (API) that it provides.

For tracking a ball, a computer vision application was developed. A color detection method was used. The application is based on OpenCV [2], which is an

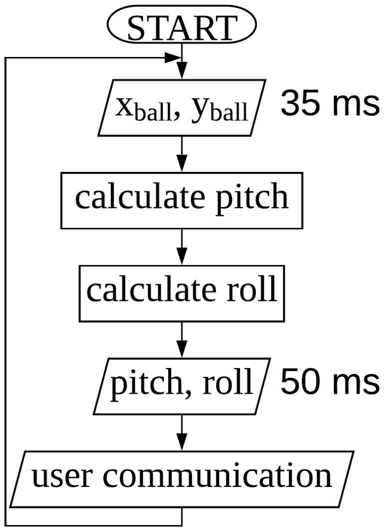

Fig. 7. A flowchart of the control application algorithm

open-source computer vision library. OpenCV puts strong focus on performance and real-time applications. Developing computer vision applications using OpenCV library is convenient because OpenCV delivers all needed subroutines and has a built-in support for range of cameras available on the market. A flowchart of the algorithm of the application is shown in Fig. 5. A vision is being obtained from a 30 frames per second (FPS) webcam. View from the camera both raw and with detected ball is presented in fig. 6.

Using a color detection method is appriopriate only in case where the detected object color is significantly different than a color of surroundings. The hue, saturation, value color model (HSV) was used. Using the HSV model eases the process of finding the boundaries values. A tool which helps choosing the color boundaries values was also developed.

All of the software is interconnected by another application which was developed. It implements a control loop. It uses subroutines provided by the software described above. It also implements proportional-integral-derivative controller (PID) and provides very simple CLI, that allows i.e. to enter PID gains. A flowchart of this application is shown in Fig. 7.

The embedded system and PC software communicate with each other using Modbus communications protocol. Modbus has become a standard communication protocol in industry. Open-source implementations of Modbus that exist for both PCs and AVR microcontrollers were used.

4. Verification Tests

Conducted verifications tests checked: 1) the accuracy of positionig the SGP, and 2) ball & plate system quality of control.

Testing an accuracy of positioning the SGP involved three test cases which test variety of pitch and roll combinations. Results of one out of three test cases are shown in Figs. 8-9. The results of the remaining two test cases are presented in tabs. 1-2.

Testing the ball & plate control system was conducted with trajectory of reference values. The measured ball position were downloaded from ball vision application. The results are shown in figs. 10-11.

Tab. 1. SGP positioning accuracy test number 2

Pitch Ref.	Roll Ref.	Pitch m.	Roll m.
0.0	-25.0	1.0	-23.0
0.0	-23.0	0.5	-20.0
0.0	-21.0	0.5	-19.0
0.0	-19.0	0.5	-16.0
0.0	-17.0	0.5	-15.5
0.0	-15.0	0.5	-13.0
0.0	-13.0	0.5	-12.0
0.0	-11.0	1.0	-10.0
0.0	-9.0	0.5	-9.0
0.0	-7.0	1.0	-4.5
0.0	-5.0	1.0	-4.0
0.0	-3.0	1.0	-0.5
0.0	-1.0	1.0	0.5
0.0	1.0	1.0	0.5
0.0	3.0	0.5	3.5
0.0	5.0	0.5	5.5
0.0	7.0	1.0	7.0
0.0	9.0	0.5	9.5
0.0	11.0	1.0	11.0
0.0	13.0	0.5	13.5
0.0	15.0	1.0	15.0
0.0	17.0	1.0	19.0
0.0	19.0	0.5	20.0
0.0	21.0	0.5	21.0
0.0	23.0	0.5	24.5
0.0	25.0	0.5	25.0

5. Conclusion

The SGP has been realized and programmed. The SGP has been effectively used in the ball & plate control system. The obtained test results proved a good quality of control.

According to the project division presented in [6], it was evaluated how much time was spent on each module. The time allocation is presented in tab. 3.

The results presented in tab. 3 show that the most cost effective tasks were related to assembly (35%) and 3D CAD model (25%).

The research works for further improvement are defined as follows:

- Reimplementation of embedded software to provide handling of real values of pitch, roll, and Δz (now it only supports integer values).
- Optimisation of all software to provide better performance in time domain.
- Use of faster camera - more than 30 FPS.

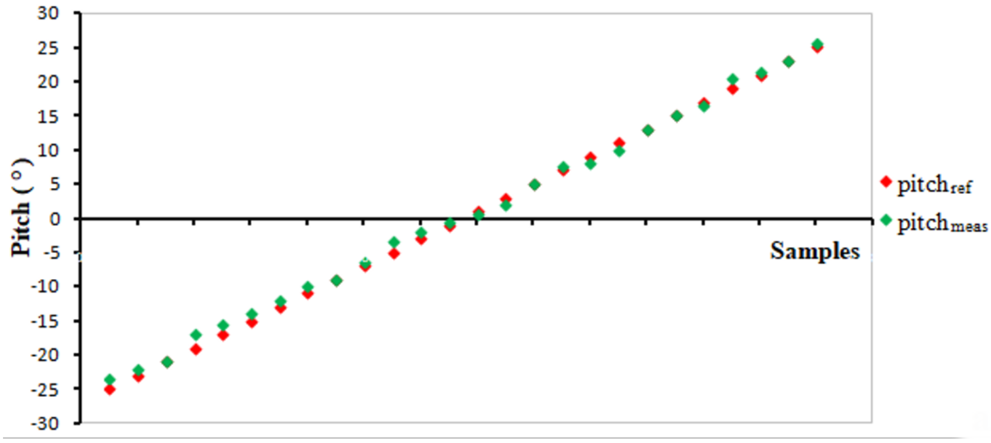

Fig. 8. Accuracy of positioning the SGP (pitch)

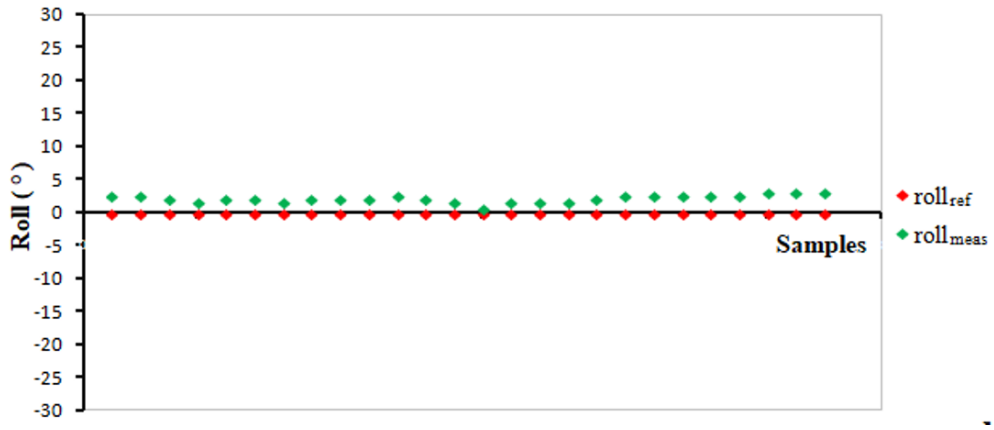

Fig. 9. Accuracy of positioning the SGP (roll)

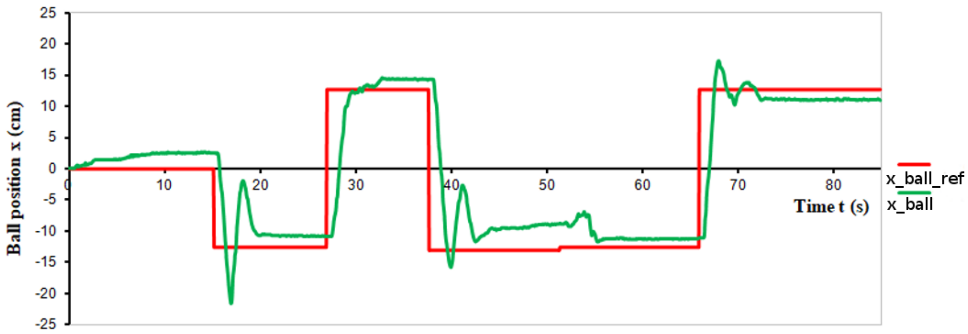

Fig. 10. Response of of ball & plate system reference trajectory (x-axis)

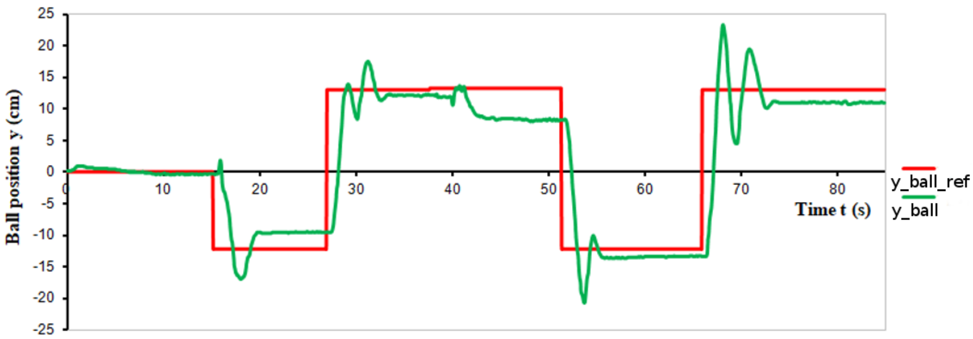

Fig. 11. Response of of ball & plate system reference trajectory (y-axis)

- Use of signal filtration.
- Further tuning of PID controllers.

Tab. 2. SGP positioning accuracy test number 3

pitch ref.	roll ref.	pitch m.	roll m.
-25.0	-25.0	-20.0	-20.0
-23.0	-23.0	-20.0	-20.0
-21.0	-21.0	-18.0	-18.0
-19.0	-19.0	-15.0	-16.0
-17.0	-17.0	-14.5	-14.5
-15.0	-15.0	-12.0	-12.5
-13.0	-13.0	-11.0	-11.5
-11.0	-11.0	-9.5	-9.5
-9.0	-9.0	-8.0	-7.5
-7.0	-7.0	-6.0	-4.0
-5.0	-5.0	-3.0	-2.0
-3.0	-3.0	-2.0	0.5
-1.0	-1.0	0.5	1.0
1.0	1.0	1.0	1.5
3.0	3.0	3.0	3.5
5.0	5.0	6.0	6.5
7.0	7.0	7.5	8.5
9.0	9.0	9.0	10.0
11.0	11.0	11.0	13.0
13.0	13.0	14.5	14.0
15.0	15.0	16.0	16.0
17.0	17.0	18.0	19.0
19.0	19.0	21.0	20.5
21.0	21.0	23.5	22.0
23.0	23.0	25.5	25.0
25.0	25.0	27.0	26.5

Tab. 3. Division of time spent on each module

Module	Time spent
3D CAD model	25%
electronics	10%
SGP positioning system	10%
ball & plate control system	10%
computer vision	10%
assembly	35%
	100%

AUTHORS

Dawid Owoc* – Gdansk University of Technology Faculty of Electrical and Control Engineering, Narutowicza 11/12, 80-233 Gdansk, Poland, e-mail: dawidowoc6@gmail.com.

Krzysztof Ludwiczak – Gdansk University of Technology Faculty of Electrical and Control Engineering, Narutowicza 11/12, 80-233 Gdansk, Poland, e-mail: krzysztof_ludwiczak07@wp.pl.

Robert Piotrowski – Gdansk University of Technology Faculty of Electrical and Control Engineering, Narutowicza 11/12, 80-233 Gdansk, Poland, e-mail: robert.piotrowski@pg.edu.pl.

*Corresponding author

REFERENCES

[1] H. Bang and Y. S. Lee, "Implementation of a Ball and Plate Control System Using Sliding Mode Control", *IEEE Access*, vol. 6, 2018, 32401–32408, 10.1109/ACCESS.2018.2838544.

[2] G. R. Bradski and A. Kaehler, *Learning OpenCV: computer vision with the OpenCV library*, Software that sees, O'Reilly: Beijing, 2011.

[3] V. E. Gough, "Contribution to discussion of papers on research in automobile stability, control and tyre performance", *Proc. of Auto Div. Inst. Mech. Eng.*, vol. 171, 1957, 392–395.

[4] K. Martin and B. Hoffman, *Mastering CMake: A Cross-Platform Build System*, Kitware, Inc.: Clifton Park, NY, 2008.

[5] P. Munford and P. Normand, *Mastering Autodesk Inventor 2016 and Autodesk Inventor LT 2016*, John Wiley & Sons, Inc: Indianapolis, Indiana, 2016.

[6] D. Owoc, K. Ludwiczak, and R. Piotrowski, "Mechatronics Design, Modelling and Controlling of the Stewart-Gough Platform". In: *2019 24th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2019, 76–80, 10.1109/MMAR.2019.8864694.

[7] M. Scarpino, *Designing Circuit Boards With EAGLE: Make High-Quality PCBs at Low Cost*, Prentice Hall: Upper Saddle River, NJ, 2014.

[8] D. Stewart, "A Platform with Six Degrees of Freedom", *Proceedings of the Institution of Mechanical Engineers*, 2016, 10.1243/PIME_PROC_1965_180_029_02, Sage UK: London, England.