

Teaching Robotics with Robot Operating System (ROS): A Behavior Model Perspective

Martin Cooney, Can Yang, Abhilash Padi Siva, Sanjana Arunesh, and Jennifer David

Halmstad University, PO Box 823, Kristian IV:s väg 3, 30118, Sweden
martin.daniel.cooney@gmail.com

Abstract. Robotics skills are in high demand, but learning robotics can be difficult due to the wide range of required knowledge, increasingly complex and diverse platforms, and components requiring dedicated software. One way to mitigate such problems is by utilizing a standard framework such as Robot Operating System (ROS), which facilitates development through the reuse of open-source code—however this also raises a challenge, in that learning curves can be steep for students who are first-time users. In the current paper, we suggest the use of a behavior model to structure the learning of complex frameworks like ROS in an engaging way. A practical example is provided, of integrating ROS into a robotics course called the “Design of Embedded and Intelligent Systems” (DEIS), along with feedback suggesting that some students responded positively to learning experiences enabled by our approach. Furthermore, some course materials, videos, and code have been made available online, which we hope might provide useful insights.

Keywords: Robotics Teaching, ROS, Behavior Model

1 Introduction: ROS-Based teaching of Robotics

The current paper reports on some of our recent experiences with teaching robotics at the postgraduate university level through Robot Operating System (ROS), leveraging a behavior model to encourage learning.

Robotics is a popular subject for which an explosion in applicability and demand is occurring [1]; but, robotics can also be demanding to learn, in encompassing knowledge in mechanics, electronics, statistics, arts, and software [2]. For software, to avoid “reinventing the wheel”, robotics practitioners can use ROS, a standard framework which offers support for typical robotics capabilities such as interprocess communication and navigation¹. One challenge with frameworks such as ROS is usability [3]: in particular steep learning curves facing first time users [4]. In conjunction with the vast amount of material which must be typically covered in learning robotics, this can dissuade teachers from incorporating such frameworks into their courses.

To facilitate the study of robotics using ROS, we suggest considering knowledge from human science, specifically behavior models, as a way to effectively structure

¹ <http://www.ros.org/>

learning experiences. From this perspective, we report on some of our experiences, positive and negative, in adding ROS to an existing robotics course, and provide some reference materials online (some course materials², videos³, and code⁴), in the hope that they might be useful for other educators.

2 Design: Structuring Course Content via Behavior Models

One important factor which has been identified as facilitating learning of challenging material is engagement [5]. Educators have sought to increase engagement in various ways, such as by seeking to foster active learning of meaningful topics and adequate support systems which allow students to feel a sense of membership [6]. In the current paper, we turned our attention to a behavior model proposed by Fogg, whose usefulness has been described for various applications (e.g., persuading people to use social media) [7].

The Fogg Behavior Model highlights the importance of three aspects in facilitating behaviors (referred to here as requirements R1-3): motivation, ability and triggers. Students should feel motivated to learn (R1). Learning challenges should reflect students' abilities (R2). Furthermore, students require opportunities to engage in learning (R3). To address these requirements we adopted an approach comprising three facets (hereafter referred to as A1-3), as depicted in Fig. 1: demonstrations, classes, and independent project work.

Demonstrations can show what positive possibilities exist, thereby eliciting pleasure or hope, or suggest how negative outcomes can be avoided, e.g., as in so-called "fear appeals" [8] (A1, addressing R1). Lectures and labs can be used to scaffold core learning, by first dedicating sufficient time to considering simplified standardized concepts and tasks, thereby promoting a perception of self-efficacy (A2, addressing R2). Project work promotes autonomy, giving students a chance to make knowledge their own by using it in practice and making decisions (A3, addressing R3).

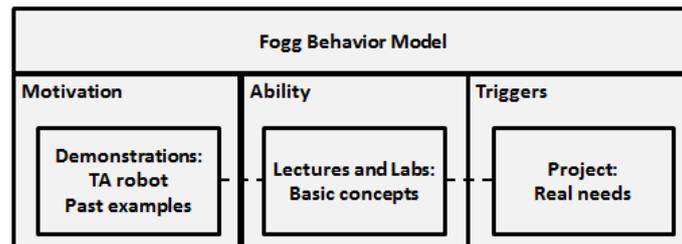


Fig. 1. Applying the Fogg Behavior Model to identify some desired components of a robotics course to facilitate incorporation and learning of complex tools such as robotics frameworks.

² http://islab.bh.se/mediawiki/images/7/72/Deis_course_description_2017.zip

³ <https://www.youtube.com/watch?v=B8nvrw7IjCI>

⁴ <https://github.com/martincooney/BaxterDemo/>

3 Implementation: The “Design of Embedded and Intelligent Systems” (DEIS) course

Informed by the behavior model, our approach was taken into account in adding ROS to an existing robotics course, “DEIS”, which is a half-year intensive (double-credit) compulsory course targeting second year master’s students at our university in Sweden. In accordance with the Bologna Process used in most European universities [9], a formal statement of examinable learning outcomes was made available. The core learning outcome is that the students should be able to improve both the breadth and depth of their conceptual and practical robotics knowledge in a collaborative, creative, and critical manner.

3.1 A1: Demonstrations

We conducted some demonstrations seeking to engage students to learn, which had not been performed in the previous year. Before the DEIS course started, students took an introductory course to robotics which did not use ROS. On the last day of this course, it was demonstrated how ROS can be used to avoid some of the challenges the students had faced (e.g., difficulty interfacing programs written in different programming languages).

Additionally, in the first three weeks at the start of the DEIS course, a robotic “teaching assistant” was introduced to show students an example of what positive things can be done with ROS, as shown in Fig. 2. This robot, composed of a Baxter humanoid upper body⁵ attached to a Ridgeback mobile base⁶, demonstrated abilities such as reading quizzes, speech and face recognition, and handing out materials. Some code for this robot has been uploaded to the internet, and details will be discussed separately [10].



Fig. 2. A robot teaching assistant was used in the DEIS course to demonstrate some robotic tasks which can be accomplished by using ROS, toward engaging students.

⁵ <http://www.rethinkrobotics.com/>

⁶ <https://www.clearpathrobotics.com/>

3.2 A2: Lectures and Labs

Content for the main part of the course supporting students' abilities to learn, the lectures and labs, was mostly retained from the previous year, while adding some material related to ROS. In total, eight teachers covered a wide range of topics, comprising statistical inference, sensors and actuators, sensor fusion, embedded programming, motion planning, simulations, communication, and image processing. In one of the first lectures, we described some of the merits and demerits of using ROS (e.g., the large community and useful tools, vs. the complexity and official support only for Ubuntu); we also defined some typical concepts (e.g., node, package, publisher, subscriber), listed some commands and tools (e.g., `catkin_make`, `rostopic list`, `roscore`; MoveIt!, Gazebo), and provided some "Hello world" examples in C++ and Python. In a follow-up lab, the students were asked to create catkin workspaces and use the ROS talker/listener tutorial to communicate between robots. Time spent for A2 was similar to that for A1. Concepts and tasks were kept simple to allow the students to perceive high ability.

3.3 A3: Project

Basic Concept. Students were also given a problem-solving project to work on in small groups, about platooning robots, which formed the core opportunity for learning robotics with ROS. The project topic, like in the previous year, was generally inspired by the Grand Cooperative Driving Challenge (GCDC), an international contest held between university teams, in which our university placed first in 2016⁷. Furthermore we focused on a scenario of cleaning, which we felt would have practical uses: For example, platoons of snow machines or snow plows are used at some ski resorts and on roads to remove snow. After disasters such as earthquakes, fires, floods, or landslides, teams cooperate to remove debris. Multiple lawn mower robots could remove grass from large open areas such as golf courses or the sides of highways, and vacuum cleaner robot teams could clean large venues such as sports arenas or hotels. Thus, we felt that such a scenario would offer various challenges and engaging opportunities to trigger learning. The main difference with the previous year, aside from the cleaning scenario, was the incorporation of ROS as a required component for the robots and infrastructure. We estimate that, although times were not recorded, the students spent more time on A3 than on A1 and A2, due to the importance of the trigger facet in allowing opportunities to make knowledge their own.

Learning environment. To implement platooning robots, the students worked in five groups, with one robot per group, in a 7.2 x 10.8m project room (80m² area) with a set up shown in Fig. 3. The robots ran on top of a 2.5 x 3.7 x 0.8m table with 0.3m walls in the middle of the room. The table was intended to be easy for students to work with robots without having to bend down, to keep robots' wheels from becoming dirty, and to stand at a desired distance (2.5m) from the ceiling to ensure that an overhead

⁷ <http://www.gcdc.net/en/>

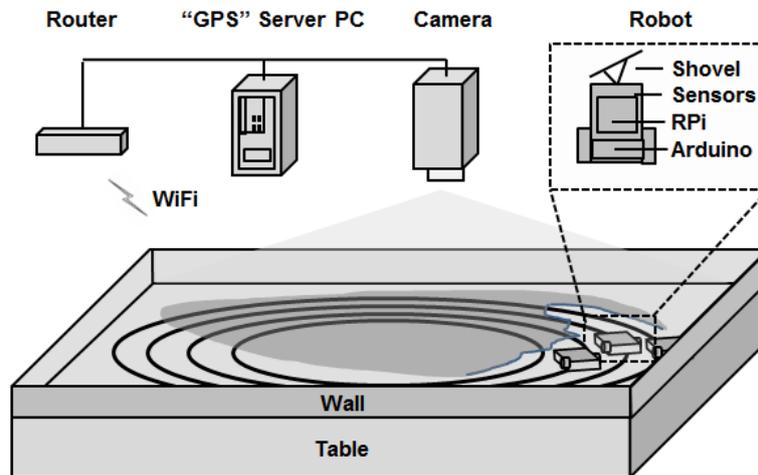


Fig. 3. Project set-up.

network camera (with 3 megapixel resolution at 20 fps, day and night) could capture the entire surface of the table. Concentric elliptical rings were added with black electrical tape to the table-top by the class (students and teachers), as tracks for the robots to run on. The room was well-lit with nine windows, and also featured 13 computers, with monitors equipped with HDMI cables to be able to also work with small computers on the robots; outside the project room was an area with tools such as 3D printers and soldering irons.

Robots. Inside the project room, students assembled and augmented some small differential drive robots from a commercially available kit using an Arduino Uno microcontroller. Sensors included an array of three line following sensors for following tracks, wheel encoders, an accelerometer, and mechanical bumpers; actuators included two 140 rpm gearmotors attached to 65mm rubber wheels, and a buzzer. After assembly, students added some additional components: a small single-board computer with in-built WiFi (Raspberry Pi 3, hereafter RPi, running a Linux operating system, Raspbian Jessie), and an 8-megapixel camera supporting 1080p30. An overhead camera, in conjunction with markers attached to the tops of robots, was also used to detect robot locations and identities.

Thus, the restriction on students was the general project theme and infrastructure, comprising a base platform using a RPi and Arduino and overhead camera. Students were also encouraged to be creative in designing their systems' appearances and capabilities. For example, the students freely selected extra components such as infrared range sensors, sonars, and servo motors to add creative features, and fashioned 3D

printed connectors and shovels for their robots. They could use school desktops, their own computers, or rely entirely on microcomputers and microcontrollers for processing. Students selected and set up power solutions using, for example, lithium polymer batteries and voltage regulators. Each group was also free to develop algorithms for detection; although in general groups started by finding colors and contours, and moved on to a more robust approach of using rotation and scale invariant log spiral markers [11]. During the course, students' choices, and their progress, were continuously monitored via a series of "tollgates", comprising reports on system design choices, presentations by individuals about topics of personal interest (the "research step"), and demonstrations of robot behavior such as lane changing.

ROS. To communicate between robots and with the overhead camera, ROS was used. All robots in the course used ROS Indigo; RPIs had the minimal ROS-Comm variant of ROS installed which features basic communication libraries but not GUI tools. Additionally, a server program was also set up for the students on a "GPS" server PC to stream images from the overhead camera; images were used to estimate the x and y coordinate positions of robots, like how automotive navigation systems can use the real Global Positioning System to estimate their positions (in other words, real GPS was not used, but we referred to the system as "GPS" due to its analogous function in allowing localization for navigation).

Within groups, one member was also designated to be a representative who would be responsible for communication and meet with the other groups to decide on a shared protocol. Teachers did not interfere in this process. It would have been possible to define a protocol for the students to use, but the choice was given to the students as a chance to foster creative thinking via problem-solving. This resulted in a set-up with four channels as shown in Table 1, allowing behaviors like in Fig. 4.

Table 1. Communication channels.

No.	Channel	Purpose
1	Heartbeat	Used by each of the robot to indicate its position in coordinates
2	Platoon Position	Used to indicate relative order (the platoon leader was -1)
3	Fan out	Used to move between a single file and diagonal formation (for traveling or cleaning respectively). Messages could also be sent from infrastructure such as an outside computer. Upon receiving a command, the leader sent commands to its followers in the platoon to change lanes.
4	Lane Change	Used when a command is sent to the "fan out channel" to send instructions to each robot, or when obstacles are detected.

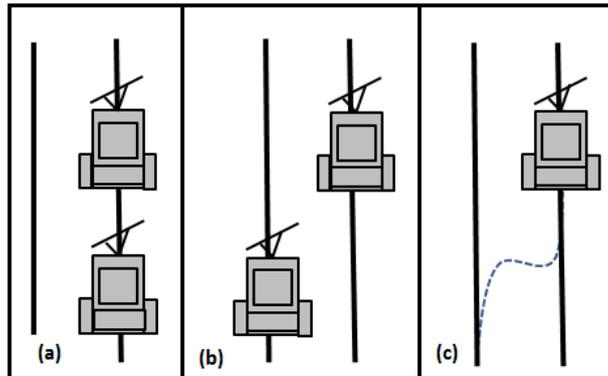


Fig. 4. Platooning examples: (a) straight line formation for traveling, (b) diagonal formation for cleaning, (c) changing lanes

The first two channels support messages which are in the style of Cooperative Awareness Messages (CAM), and the latter two, in describing traffic events, resemble Decentralized Environmental Notification Messages (DENM) [12]. Thus, ROS was used to enable some platooning behaviors in the project robots.

3.4 Examination

Overall, as in the previous year, students were graded 50% based on conceptual knowledge and 50% based on practical knowledge, through an oral exam and written report respectively. Criteria were as follows:

- Grade U (Fail): Basic requirements not met
- Grade 3: The student demonstrated collaboration, to apply basic concepts
- Grade 4: + Creativity
- Grade 5: + Critical thinking/excellent methodology

This year, 24 students (average age = 26.8 years, SD = 4.7, 8 female, 16 male) participated.

4 Experiences and Discussion

We gained some feedback on our course using ROS, in the final demonstration of the course project, through an anonymous survey conducted by our school, and by asking students. We note that in teaching it is generally difficult to conduct rigorous evaluations controlling only a single facet such as the usage of ROS, as there are typically many lessons learned throughout a course and many changes made toward allowing for the best possible learning experiences. Nonetheless we also present some comparison with feedback from the previous year when ROS had not been incorporated, in the estimation that the general trends can be informative.

4.1 Final Demonstration

In this year's course with ROS, students were able to develop additional functionality when compared to the previous year: moving in diagonal formation, avoiding obstacles, and clearing debris. Fig. 5 shows some scenes from the final demonstration of students' work with ROS, and some videos have been made available online⁸.

4.2 Survey

Additionally, a survey was conducted by our university, obtaining feedback from 9 participants in regard to the questionnaire items below (using a six point scale, where 0 meant strongly disagree, and 5 meant strongly agree):

- The design of the course (teaching and examinations, etc.) has enabled me to attain the learning outcomes of the course.
- The content of the course (required reading, lectures, etc.) has enabled me to attain the learning outcomes of the course.
- Through the course, I was able to take part in research relevant for the field.
- Through the course, I developed my ability for critical thinking.
- The course encouraged me to actively search for and acquire new knowledge/abilities/skills within the field.

The average result was 4.0 (80%), which was an improvement from the previous year's score when ROS was not used, 3.5 (70%). Students also described some positive and negative experiences. Over half of the respondents described the project using ROS as the most worthwhile element in the course, with one mentioning the robot teaching assistant; this represented an increase from the previous year in which only two students mentioned the project. In terms of improvement, students suggested the course design could be structured to allow further freedom to select topics of interest, and that platooning had been difficult because it was hard to find times to share robots with other groups, among other comments (e.g., that the course room could be larger and that better hardware could be helpful).

4.3 Additional feedback

The survey yielded some useful information but did not specifically relate to ROS; therefore, the students were also asked for feedback about any problems they had experienced with ROS. Familiarization with basic concepts and installations were described as time-consuming, such as installation of the `cv_bridge` package on the RPis, or various versions of OpenCV. Delays were also reported as a problem. One example referred to synchronization with Matlab for image processing while using many nodes. Another example reported not knowing how to select a preferred protocol for messages: e.g., assuming latency could be more important than reliability for heartbeat messages, UDPROS could have been used instead of TCPROS. As well,

⁸ <https://www.youtube.com/watch?v=B8nvrw7IjCI>

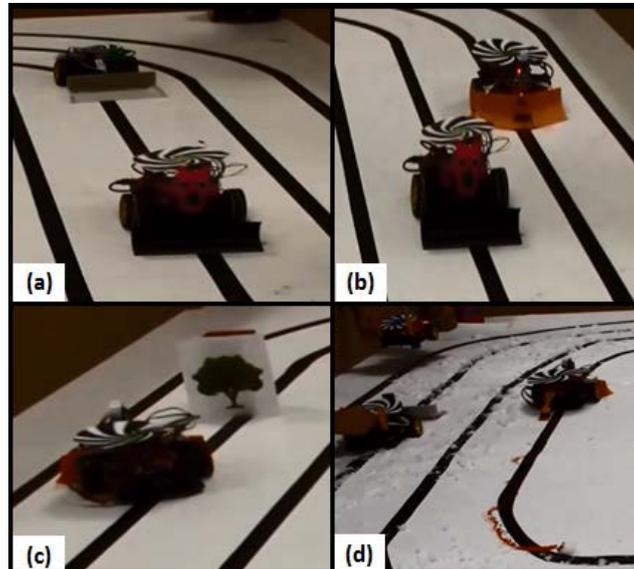


Fig. 5. Results of ROS-based project work: (a) Platooning in a straight line formation to reach a destination, (b) Diagonal formation to clean, (c) Swerving out of the way of a detected obstacle (here a “tree”), and (d) a final task in which robots had to deal with some artificial “snow” (movements were visualized by attaching a paint brush to the backs of the robots).

some tasks like line-following did not involve ROS, so some students, especially the communication “representatives” in each group, received more time to work with ROS than others. Despite such considerations, a number of students also voiced positive comments about their experiences.

4.4 Conclusions

In summary, we observed that incorporating ROS into an existing robotics course, guided by considering a behavior model, appeared to have allowed students to develop more capabilities with their robots, and feedback from students was more positive than in the previous year. As a result, we have decided to continue to use ROS in our course. Next year, we will consider how to further incorporate ROS functionalities, such as *rosserial* for the robots’ microcontrollers or *bag* files for sensor data, and also to allow more freedom for students to explore topics on their own. We also plan to take into account lessons learned in the current year; for example, each group will receive two robots instead of one, which will let more students get hands-on with ROS, toward achieving more effective learning.

We expect that effective learning of ROS in university courses, also leveraging behavior models and knowledge of how to engage students, will contribute to robotics

in both academia and industry, as students bring their knowledge and engagement with them to new endeavors.

5 Acknowledgements

We would like to thank everyone who helped, including the students of the DEIS course! The authors received funding from the Swedish Knowledge Foundation (Sidus AIR no. 20140220 and CAISR 2010/0271).

References

1. Pratt, G. A.: Is a Cambrian Explosion Coming for Robotics? *Journal of Economic Perspectives*, 29(3): 51-60 (2015). DOI: 10.1257/jep.29.3.51
2. King, K. P., Gura, M. (Eds.): *Classroom robotics: Case stories of 21st century instruction for millennial students*. IAP, 117 (2007).
3. Toris, R., Kammerl, J., Lu, D. V., Lee, J., Jenkins, O. C., Osentoski, S., et al.: Robot web tools: Efficient messaging for cloud robotics. In: *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 4530-4537. IEEE (2015).
4. Coleman, D., Sucan, I., Chitta, S., Correll, N.: Reducing the barrier to entry of complex robotic software: a MoveIt! case study. *arXiv preprint arXiv:1404.3785* (2014).
5. Hake, R. R.: Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. *Am. J. Phys.* Vol. 66, 64-74 (1997).
6. Marks, H. M.: Student engagement in instructional activity: Patterns in the elementary, middle, and high school years. *American educational research journal*, 37(1), 153-184 (2000).
7. Fogg, B. J.: A Behavior Model for Persuasive Design. In: *Proceedings of the 4th International Conference on Persuasive Technology*. Persuasive '09. New York, NY, US: ACM: pp. 40:1-40:7 (2009). doi:10.1145/1541948.1541999. ISBN 9781605583761.
8. Maddux, J. E., Rogers, R. W.: Protection motivation and self-efficacy: A revised theory of fear appeals and attitude change. *Journal of experimental social psychology*, 19(5), 469-479 (1983).
9. Wächter, B.: The Bologna Process: developments and prospects. *European journal of education*, 39(3), 265-273 (2004).
10. Cooney M., Leister W.: *Designing an Engaging Robotic Teaching Assistant Using the Engagement Profile*. (2018) (working paper).
11. Karlsson S., Bigun J.: Synthesis and detection of log-spiral codes. In: *SSBA Symposium i bildanalys*, pp. 4 (2011).
12. Santa J., Pereniguez F., Moragón A., Skarmeta A. F.: Vehicle-to-infrastructure messaging proposal based on CAM/DENM specifications. In: *Wireless Days (WD), 2013 IFIP* pp. 1-7. IEEE (2013).