Development of a Digital Twin for a Mobile Articulated Gripper Robot in Simscape Multibody

Andreas Apostolatos[1,2*], Jan Janse van Rensburg[1], Sebastian Groß[1], Stefan Kerber[1], Claudia Rubio Hervás[2], Minas Apostolakis[2], Chi-Chia Tung[2], Miguel A. Vega Torres[2], Steve Miller[1]

[1]The MathWorks GmbH, Weihenstephaner Straße 6, 81673 Munich, Bavaria, Germany; *aapostol@mathworks.com, andreas.apostolatos@tum.de

[2]Technical University of Munich, Arcisstraße 21, 80333 Munich, Bavaria, Germany

SNE 35(2), 2025, 83-91, DOI: 10.11128/sne.35.tn.10732
Selected ASIM SST 2024 Postconf. Publication: 2024-12-10
Received Improved: 2025-03-11; Accepted: 2025-03-20
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Mobile articulated gripper arms are revolutionizing a range of fields, from additive manufacturing and packaging to automated assembly lines and surgical robotics. These innovations underscore the crucial role of simulation in design and development. Digital Twins are integral to this process, ensuring robustness, performance optimization, real-time monitoring and control, and adherence to industry standards. Preparing students for successful careers to develop such innovative engineering solutions leveraging state-of-the-art methods and approaches is of high importance. In this work, we demonstrate how *Project-Based Learning* (PBL) for such complex engineering tasks can be significantly enhanced and accelerated using MATLAB® and Simscape™ Multibody™. The results of this project clearly show that students can deliver high-end, robust solutions to complex engineering tasks, allowing them to efficiently familiarize themselves with advanced engineering topics by leveraging industry-mature modeling, simulation, test, and deployment ecosystems provided by MathWorks®.

Introduction

Today's job market requires highly skilled engineers equipped with the knowledge and tools to elaborate, develop, test, and eventually deploy qualified engineering solutions for diverse applications.

As new technologies continuously emerge, the need for sophisticated development methods and comprehensive environments increases, enabling the handling of ever broadening application fields. Therefore, modern education necessitates the adoption of effective and efficient teaching methods such as *Flipped Classroom* and *Project-Based Learning* (PBL), see also in [1, 2, 3, 4]. These allow students to gain practical experience using state-of-the-art platforms and methods they will eventually encounter in industry.

Figure 1: Kinova Gen3 robotic arm [5].

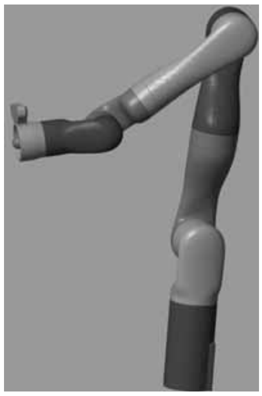

Figure 2: Kinova Gen3 robotic arm modelled in Simscape Multibody.

The *Technical University of Munich* (TUM) in Munich, Germany, is one of the leading technical academic institutes in Europe and worldwide. It offers many diverse and interdisciplinary study programs in engineering. Most these programs, if not all, involve PBL as part of their curricula.

This publication illustrates the development and investigates the results of a student project within the frame of PBL-based course "Software Lab" that took place in 2023. This lab is offered to master-level students at TUM annually. Academic and industrial partners collaborate with the participants by offering projects to pursue during this course, thus enabling students to elaborate practical projects of engineering significance in both academia and industry.

The results show that the use of PBL with industry-established engineering software increases motivation for the project elaboration, promotes learning outcomes, and enables students to develop practical and industry-relevant skills.

1 Problem Definition

This student project was conducted in collaboration with MathWorks®, the makers of MATLAB® and Simulink® modeling and simulation platforms. Simscape™ Multibody™, an add-on product to Simulink, was used as Simulation platform. Simscape Multibody enables the modeling and simulation of multibody systems, which are integral part of robotic solutions. Coupled with the Control Design and Hardware Deployment capabilities of Simulink, Simscape Multibody enables modeling, simulation, test, and deployment of complex, industrial robotic systems.

The student team's engineering task was the development of a Mobile Articulated Gripper Robotic Arm, which can be used for tasks such as additive manufacturing, surgery, search and rescue missions, etc. As starting point for the project, the Multibody model [6] of the Kinova Gen3 [5] robotic arm was provided. The actual Kinova Gen3 robotic arm is shown in Figure 1, whereas its Multibody representation in Simscape Multibody is depicted in Figure 2. It is evident that the digital model exhibits strong resemblance to its physical counterpart.

The project objectives included:

- The setup of a multibody system for the mobile base with three wheels in Simscape Multibody,
- the setup of a multibody system for the mobile base coupled with the multibody representation of the articulated Gen3 robotic arm from Kinova,
- the implementation of a path following maneuver for the mobile base,
- the implementation of a "pick and place" maneuver for the articulated robotic arm using inverse kinematics, and
- the coordination of the different tasks of the robot using a *Task Scheduler*, aiming to replicate an automated wall construction using Simscape Multibody.

The project goals were achieved in a timely manner with remarkable efficiency given the students' limited knowledge in robotics, MATLAB, and Simscape Multibody prior to the project elaboration.

The outcome of this study shows that MATLAB, Simulink, and Simscape Multibody form a strong platform for PBL due to their flexibility and adaptability. This enables students to fully leverage their skills while working on complex tasks.

Familiarization with the tools and methods needed for an engineering project like this is not trivial. The students were asked to use the MathWorks self-paced *Online Training Suite* (OTS) [7] to gain the required tools knowledge and skills. These self-paced online courses allow for an efficient and seamless familiarization with the MathWorks products and environments, necessary for the project elaboration.

2 Project Elaboration

The project was expected to be elaborated and finalized throughout an academic year. Regular presentations were scheduled to allow the project participants disseminate their results, share the project status, and respond to questions from their peers, the project supervisors, and the course coordinators.

2.1 Articulated robotic arm and gripper end effector

A multibody system in `urdf`-format for the Kinova Gen3 robotic arm was provided as a starting point for the project. Function `smimport` [8] from Simscape Multibody allows for converting a CAD model (or an assembly) to a multibody system in Simscape Multibody, which considerably accelerates the development process.

In practice, one would employ torque control to move the joints of the articulated robotic arm and gripper end effector. It was suggested that motion control should be used instead to simplify the task.

This modeling approach is suitable for early-stage design. Thus, the desirable motion was simply applied to the robotic joints. This suitable for early-stage design simplification allowed the students to complete the project promptly without the need for time-consuming controller tuning. Moreover, this modeling approach is equivalent to assuming that all necessary torque can be applied to the joints to achieve the desirable joint motion.

Subsequently, a set of waypoints was defined to have all joints of the articulated robotic arm operate in a coordinated manner so that the end effector can achieve the desirable end position. Using an appropriate number of waypoints allows the joints to find unique motion paths, leveraging the *Inverse Kinematics* (IK) solver in Simscape `simscape.multibody.KinematicsSolver` [9]. Just using a handful of waypoints might not be enough for the IK solver to find unique actuations for all joints, as the inverse kinematics problem is by nature highly non-linear.

Figure 3 shows the motion of the end effector gripper moving from the initial vertical position to finally reaching the brick on the ground. This highlights the efficiency and robustness of IK solver in Simscape.

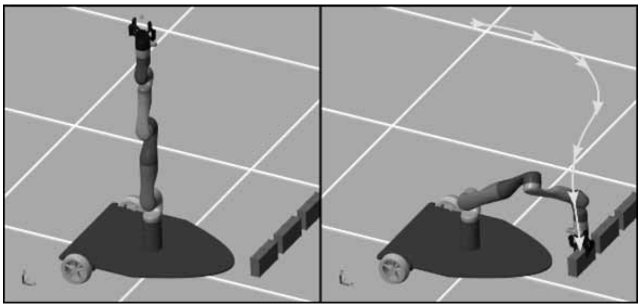

Figure 3: Inverse Kinematics Solver in Simscape (`simscape.multibody.KinematicsSolver`).

2.2 Mobile base

The students decided to use a three-wheeled model to model the mobile base. This is convenient because it provides the necessary stability and allows for steering by controlling the rotational speed of each rear wheel. Additionally, there is no need for a suspension. This design is inspired by one of the models in the SolidWorks® tutorial; see in [10] for more information.

Exporting the model from SolidWorks® to Simscape multibody can be achieved using the *Simscape Multibody Link* [11].

Various studies with different parameters were performed to find the most appropriate parameters that would allow the model to simulate stably. Moreover, the contact behavior between the ground and the wheels was modeled using the *Spatial Contact Force* block [12] in Simscape. Tables 1 and 2 summarize the values chosen for the normal and frictional contact force coefficients used in this project, respectively.

Parameter	**Value**
Method	Smooth Spring-Damper
Stiffness	1*e*6 N/m
Damping	1*e*4 N/(m/s)
Transition region width	0.3 m

Table 1: Normal contact behavior.

Parameter	**Value**
Method	Smooth Stick-Slip
Coefficient of static friction	0.9
Coefficient of dynamic friction	0.7
Critical velocity	1*e*-2 m/s

Table 2: Frictional contact behavior.

Subsequently, the students sought to impose a path the mobile base should follow. As the robotic platform under consideration is of differential drive robot type, they decided to use the *Pure Pursuit* block in Simulink, (see in [13]), to obtain the linear and angular velocity control commands necessary for navigating the mobile base via a track defined by a provided set of waypoints. The students leveraged a *MATLAB Function* block [14] to convert these linear and angular velocity control commands to the rotational speed of the left and right wheels of the mobile base. The corresponding Simulink schematic is depicted in Figure 4. The Pure Pursuit block in Simulink is provided with the robot's pose, namely, its position and heading of the robot, and the corresponding waypoints it should navigate through.

It returns the necessary control inputs for the linear and angular (rotational) velocities v and ω, respectively, that the robot must possess to navigate via the provided waypoints with the given pose.

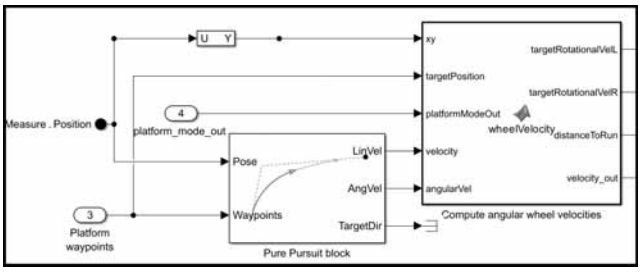

Figure 4: Pure Pursuit block and computation of the left and right wheel rotational velocities.

However, these control signals need to be converted to rotational speeds for the left and right wheel of the mobile base. To achieve that, the following formulas were employed and implemented in the MATLAB Function block with name *"Compute angular wheel velocities"*:

$$\omega_{\mathrm{L}} = \frac{v - \frac{d}{2\omega}}{r} \tag{1a}$$

$$\omega_{\mathrm{R}} = \frac{v + \frac{d}{2\omega}}{r} \tag{1b}$$

Please note that d and r in Equations 1 stand for the track width, namely, the distance between the centerline of the two rear wheels, and the radius of each rear wheel, respectively.

It is worth noting that the pure pursuit algorithm cannot stabilize the robot once it has reached its destination. For this purpose, a triangular velocity profile reduction was implemented to stabilize the behavior of the mobile base. This ensures that the mobile base reaches its destination with a gradual reduction in its velocity, ultimately stopping at the destination. A proportional controller is used to obtain the required torque that needs to be applied at the left and right wheels to achieve the desirable rotational speed. The results of the target and achieved rotational velocities for both wheels are shown in Figure 5. A three-fold scaling of the error in terms of the rotational velocity for both wheels is used for the proportional controller. This simple P-controller can almost perfectly control the torque to achieve the desirable rotational velocity.

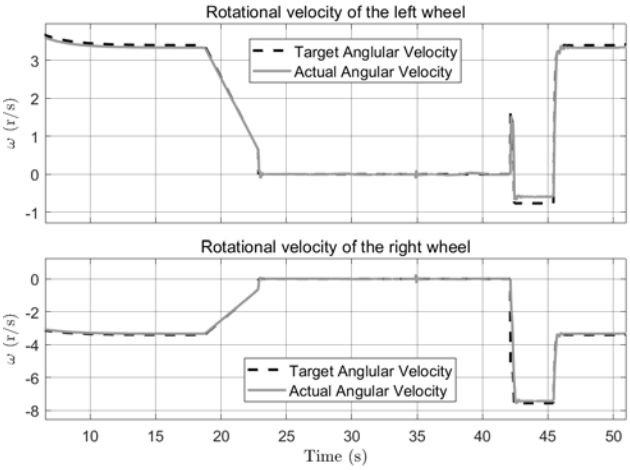

Figure 5: Target and achieved rotational velocities for both wheels of the mobile base using a proportional controller.

The triangular velocity profile employed for reducing the wheel's rotational velocity in the vicinity of the destination between can be clearly seen in Figure 5 at time instances 19 and 22.8 seconds. Employing this approach enables the mobile platform to reach its destination with stability and efficiency.

The waypoints are defined such that the mobile base navigates from the brick picking to the brick placing position. Moreover, Figure 6 depicts the track of the mobile base.

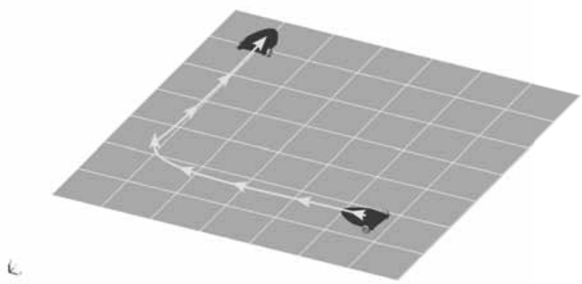

Figure 6: Mobile base navigating via waypoints using the Pure Pursuit algorithm in Simulink.

2.3 Coupling of the articulated robotic arm and the mobile base

Thus far, it has been highlighted how the students modeled and simulated the articulated robotic arm together with the gripper end effector and the mobile base separately in Simscape Multibody.

A *Weld Joint* block [15] was used to establish a rigid connection between the two; see Figure 7.

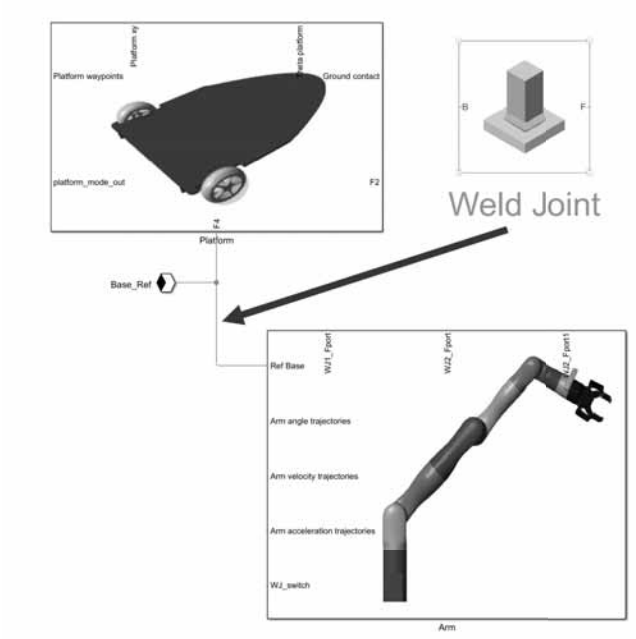

Figure 7: Rigid connection of multibody parts using the Weld Joint block in Simscape Multibody.

Joints in Simscape Multibody enable coupling different parts of a multibody system by introducing *Degrees of Freedom* (DOFs) that may restrict or enable the desirable motions depending on the application. Since the mobile base can move freely in this case, using a Weld Joint block between the mobile base of the articulated gripper arm has the desirable effect of mounting the articulated gripper arm onto the mobile base rigidly.

2.4 Power supply for the rear wheels of the mobile base

The electrical system used to power the rear wheels of the mobile base is modeled in Simscape as a DC motor. The electrical motor is connected to a battery, shown in the blue section of the Simscape model in Figure 8. The *Motor & Drive (System Level)* block [16] was used to convert the electric current from the electrical part of the Simscape model to the mechanical part (see the green part in the Simscape mode in Figure 8). Simscape offers many blocks that allow Simscape models to be interfaced from different domains (mechanical, electrical, fluid, etc.). Finally, connecting the mechanical Simscape system in green to the Simscape Multibody system is also necessary.

The torque and speed connections are denoted in Figure 8 using brown color. The *Rotational Multibody Interface* block is used [17] for this purpose.

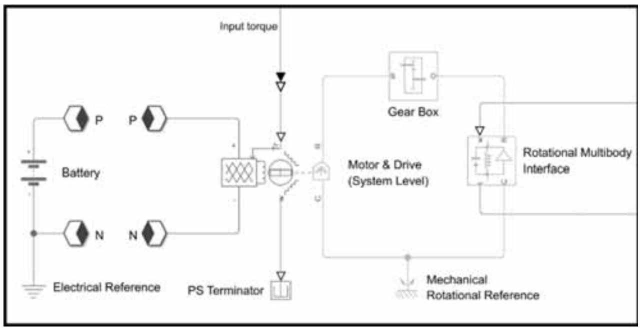

Figure 8: Modeling of a DC motor in Simscape.

The Rotational Multibody Interface block converts the mechanical force from the mechanical part of the Simscape model to torque that can be applied to the corresponding joints in the Simscape Multibody system. In this case, the torque is applied directly to the rear wheels of the mobile base. As mentioned in Section 2.2, the torque is controlled using a proportional controller by scaling the error in the rotational velocity of each of the rear wheels threefold. Each rear wheel is powered by identical DC motors, as shown in Figure 8.

2.5 Task scheduler

A task scheduler had to be implemented in the simulation model to handle task management. The mobile articulated robotic arm must drive to the location of the bricks, pick up each brick, bring the robotic arm into a vertical position (idle position) minimizing the inertia while driving, drive to the location where each brick is supposed to be placed, and finally place the brick.

The four main tasks that the task scheduler organizes include:

- Driving towards the destination where the bricks are stored,
- picking up one brick at a time,
- restoring the robot's arm posture to vertical (idle position),
- driving towards the destination where the bricks are to be placed,
- placing each brick to construct a wall.

These steps are shown in Figures 9, 10, 11, 12 respectively. The task scheduler is implemented using a MATLAB Function block by means of switch branches based on a flag. The logic of the task scheduler could be also implemented graphically using Stateflow®, but the students decided to use MATLAB code in this project instead.

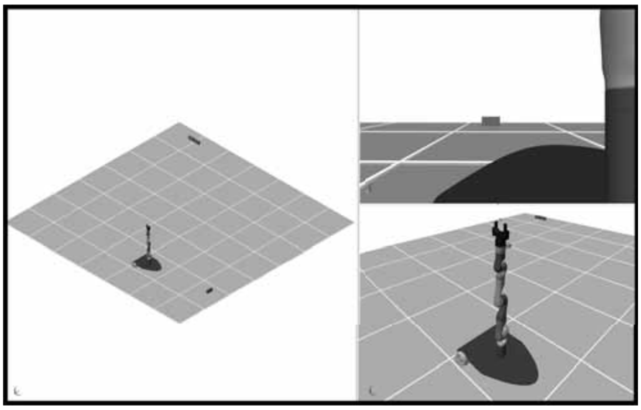

Figure 9: Task scheduler: Drive towards the destination where the bricks are stored with the arm in vertical posture (idle position).

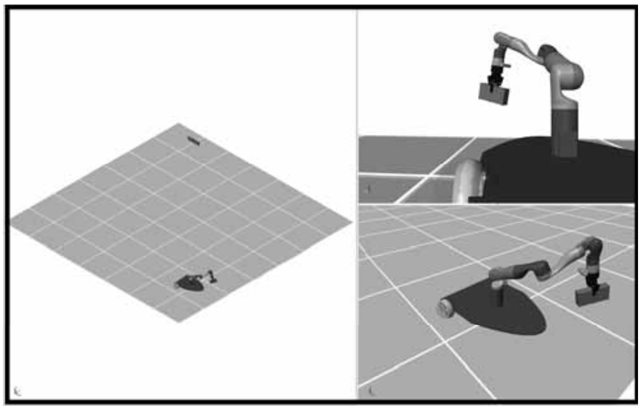

Figure 10: Task scheduler: Pick up each brick using the IK solver in Simscape.

3 Project Results and Learning Outcomes

The main project objective of modeling and simulating a mobile articulated gripper robot that can pick and place bricks to construct a wall using Simscape Multibody has been achieved with remarkable efficiency.

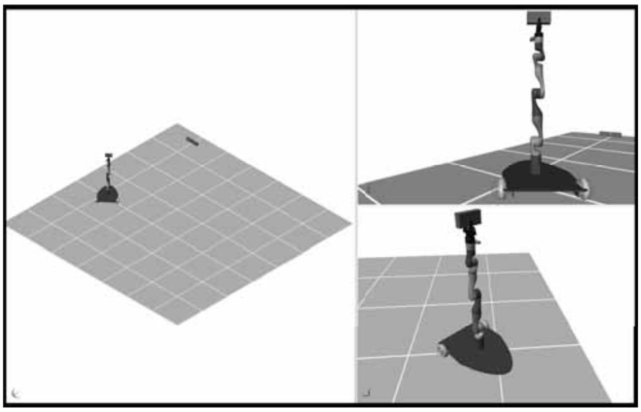

Figure 11: Task scheduler: Drive towards the destination where the bricks should be placed.

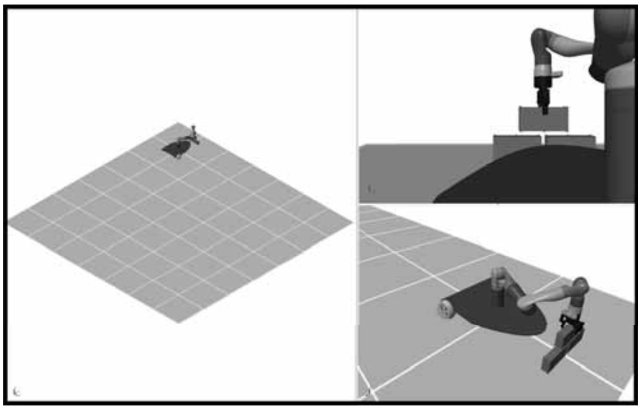

Figure 12: Task scheduler: Place bricks at the destination to construct a wall using the IK solver in Simscape.

Next, the power needed for the articulated gripper arm to pick the third, and last, brick (see Figure 13 accordingly) is investigated.

The power consumption for each joint can be computed using the following equation:

$$P = t \cdot \omega \tag{2}$$

where t and ω stand for the angle by which each wheel of the mobile base rotates and the corresponding angular (rotational) velocity. Simscape offers the *Rotational Power Sensor* block [18], among other sensor blocks, that can be leveraged to compute the power of a rotational mechanical system, see Equation 2. Figure 14 shows the physical conserving ports of this sensor block, which are physical connections that conserve the mechanical system's rotational energy and the output signal port that senses the corresponding rotational power.

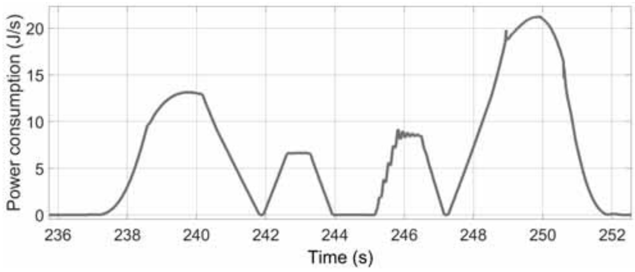

Figure 13: Power consumption of the articulated robot arm when picking up the last brick.

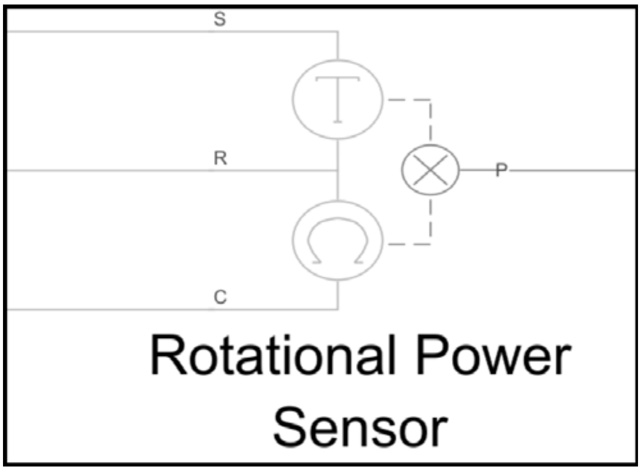

Figure 14: Power consumption of the articulated robot arm when picking up the last brick.

Joints in Simscape Multibody enable coupling different parts of a multibody system by introducing DOFs that may restrict the motions as needed for the corresponding application. Using a Weld Joint between the mobile base of the articulated gripper arm has the desirable effect of mounting the articulated gripper arm onto the mobile base rigidly since the mobile base can freely move on the ground in this case.

The total power consumption needed for the movement of the mobile base is also computed in the same manner. The results are summarized in Figure 15.

The power needed for the mobile base can be exclusively attributed to the actuation of the two rear wheels according to the pure pursuit algorithm for path following and the corresponding *P*-controller. The relatively short instances where the power consumption becomes negative are worth noting. These can be attributed to the moments when the velocity is reduced as the mobile base approaches the destination using the employed triangular velocity profile u sing t he p roportional controller.

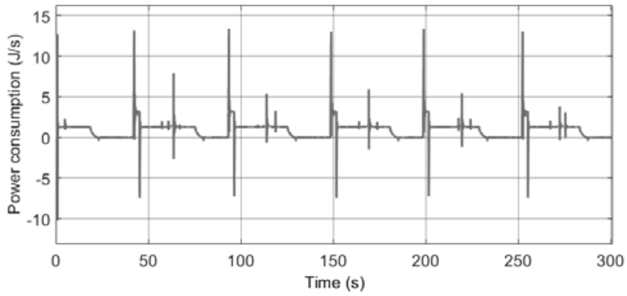

Figure 15: Power consumption of the mobile base throughout the simulation.

When the robot's velocity is reduced by motion input, kinetic energy flows back into the electric motor. These instances can be considered as equivalent to regenerative breaking.

The students were able to ramp up quickly on the underlying topics even though they weren't experts in robotics nor familiar with the computational platforms employed prior to starting the project. They felt comfortable discussing advanced topics in robotics by the end of the project, such as physical modeling, control algorithms, inverse kinematics, and gaits among other. They proposed appropriate solutions to the challenges faced throughout the project.

This project-based learning experience allowed the students to gain significant knowledge in robotics, the employed computational platforms, and team collaboration by working on a project with high practical impact.

4 Conclusions

This work underscores the importance of using adaptable and flexible simulation platforms, such as MATLAB, Simulink, and Simscape, to conduct courses based on Project-Based Learning, which enables students familiarize themselves with prominent industrial methods and workflows.

Moreover, the importance of simulation for the early-stage design of robotic systems in the context of advanced engineering applications is also highlighted herein. It emphasizes the effective and efficient modeling and simulation of robotic multibody systems using Simscape Multibody, Simulink, and MATLAB.

These tools offer the flexibility to choose the level of detail necessary for the required system fidelity.

For example, we demonstrate that while the mobile base motors are modeled with a basic DC electric circuit, the articulated robotic arm's joint motion is simulated directly without specifying the power source. This approach simplifies early-stage design processes by focusing on essential elements. Additionally, we highlight that control design can be directly integrated within Simulink, allowing for easy computation and sensing of the required power using Simscape sensing blocks.

In conclusion, simulation is essential for any stage design of such robotic systems. It provides critical insights into the model under consideration that can be used to enhance productivity, robustness, and eventually, safety when such complex engineering solutions are deployed in practical applications.

Moreover, simulations are widely used to produce data to train reinforcement learning models, create policies for optimal robot control, or train surrogate models from multibody systems via *Deep Learning* to speed up computational time, see for instance in [19].

To this end, modern teaching methods, such as Flipped Classroom, Project-Based Learning, Active Learning, etc., can be considerably enhanced by using industry-mature simulation plat-forms like the ones used in this study.

Publication Remark

This contribution is the improved version of the conference version published in
Tagungsband Langbeiträge ASIM SST 2024,
ARGESIM Report AR 47,
ISBN ebook: 978-3-903347-65-6, pp 95-102,
DOI 10.11128/arep.47.a4722

References

[1] Dimitra Kokotsaki, Victoria Menzies, and Andy Wiggins. Project-based learning: A review of the literature. *Improving schools*, 19(3):267–277, 2016.

[2] José Alberto Naves Cocota, Thiago D'Angelo, and Paulo Marcos de Barros Monteiro. A project-based learning experience in the teaching of robotics. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, 10(4):302–309, 2015.

[3] Ignacio de Los Rios, Adolfo Cazorla, José M Díaz-Puente, and José L Yagüe. Project–based learning in engineering higher education: two decades of teaching competences in real environments. *Procedia-Social and Behavioral Sciences*, 2(2):1368–1378, 2010.

[4] Andreas Apostolatos and Sebastian Gross. Teaching advanced topics in numerical engineering using project-based learning. In *2024 IEEE Global Engineering Education Conference (EDUCON)*, pages 01–10. IEEE, 2024.

[5] Kinova Robotics. Gen3 robotic arm. `https://www.kinovarobotics.com/product/gen3-robots`. Accessed: 2025-03-07.

[6] Steve Miller. Industrial robot models in simscape. GitHub, 2025. Accessed: 2025-03-07.

[7] MathWorks. Online training suite. `https://matlabacademy.mathworks.com`. Accessed: 2025-03-07.

[8] MathWorks. simimport function. `https://www.mathworks.com/help/sm/ref/smimport.html`. Accessed: 2025-03-07.

[9] MathWorks. Kinematicssolver. `https://www.mathworks.com/help/sm/ref/simscape.multibody.kinematicssolver.html`. Accessed: 2025-03-07.

[10] SolidWorks. Solidworks full tutorial – mobile car. YouTube video. `https://www.youtube.com/watch?v=U8WFaWKi3yo`, Accessed: 2025-03-07.

[11] MathWorks. Simscape multibody link. `https://www.mathworks.com/help/smlink/ug/installing-and-linking-simmechanics-link-software.html`. Accessed: 2025-03-07.

[12] MathWorks. Spatial contact force block. `https://www.mathworks.com/help/sm/ref/spatialcontactforce.html`. Accessed: 2025-03-07.

[13] MathWorks. Pure pursuit block. `https://www.mathworks.com/help/robotics/ref/purepursuit.html`. Accessed: 2025-03-07.

[14] MathWorks. Matlab function. https://de.mathworks.com/help/simulink/slref/matlabfunction.html. Accessed: 2025-03-07.

[15] MathWorks. Weld joint. https://www.mathworks.com/help/sm/ref/weldjoint.html. Accessed: 2025-03-07.

[16] MathWorks. Motor & drive (system level). https://www.mathworks.com/help/sps/ref/motordrivesystemlevel.html. Accessed: 2025-03-07.

[17] MathWorks. Rotational multibody interface. https://www.mathworks.com/help/simscape/ref/rotationalmultibodyinterface.html. Accessed: 2025-03-07.

[18] MathWorks. Rotational power sensor. https://www.mathworks.com/help/sdl/ref/rotationalpowersensor.html. Accessed: 2025-03-07.

[19] MathWorks. Twt gmbh develops new workflow for tuning automotive suspension designs using deep learning and multibody simulation. https://de.mathworks.com/company/user_stories/twt-gmbh-develops-new-workflow-for-tuning-automotive-suspension-designs-using-deep-learning-and-multibody-simulation.html. Accessed: 2025-03-07.