Fumiya Iida · Perla Maiolino ·
Arsen Abdulali · Mingfeng Wang (Eds.)

Towards Autonomous Robotic Systems

24th Annual Conference, TAROS 2023
Cambridge, UK, September 13–15, 2023
Proceedings

Extended Abstracts

POSTER Session 1.1

Lecture Theatre 1 (LT1)

PRESENTER	POSTER
David BATTY	Occupancy Map Abstraction for Autonomous Ground Rover Exploration in Hazardous Environments
Daniel S. JOHNSON	Receding Horizon Contact Planning for Advanced Motions in Hexapod Robots
Roee M. FRANCOS	Spiral Sweeping Protocols for Detection of Smart Evaders
Christopher WHEAT	Detection and Classification of Unmanned Aerial Vehicles using Commercial Off-The-Shelf Radar Detectors
Qi ZHANG	Towards Lifelong Social Robot Navigation in Dynamic Environments
Konstantin YAKOVLEV	Evaluation of Safety Constraints in Autonomous Navigation with Deep Reinforcement Learning
Md Nazmul HUDA & Sarfraz AHMED	Investigation of Action Recognition for Improving Pedestrian Intent Prediction
Gjosse ZIJLSTRA	Toward Semi-Autonomous Terrestrial Robots for Atmospheric Electricity Measurement
Md Nazmul HUDA & Zhiyuan Yang	Evaluation of SLAM algorithms for Search and Rescue applications
Mal FAZLIU	Automating Robotic Gas Distribution Mapping in Unknown and GPS-denied Environments
Metin OZKAN	Developing an Integrated Runtime Verification for Safety and Security of Industrial Robot Inspection System

POSTER Session 1.2

Lecture Theatre 2 (LT2)

PRESENTER	POSTER
Elijah ALABI	Evaluation Of OSMC Open Source Motor Driver for Reproducible Robotics Research
Lupo MANES	Cable-loop Gripper Add-on for Multimodal Grasping of Lab Supplies
Dexter SHEPHERD	Low-resolution Sensing for Sim-to-real Complex Terrain Robots
Kshitij GAIKWAD	Open Source Hardware Robotics Interfacing Board
Adam SPIERS & Jayjun LEE	Naturalistic Robot Arm Trajectory Generation via Representation Learning
Adam SPIERS & Ciaran DOWDS	Object Feature Reconstruction via Robotic In-Hand-Manipulation and Haptic Proprioception
Ian HOWARD	Design and kinematic analysis of a 3D-printed 3DOF robotic manipulandum
Shibao YANG	Reinforcement Learning-based Adaptive Probabilistic Movement Primitives in Hybrid Scenarios (Extended Abstract)
Gautham DAS & Laurence ROBERTS-ELLIOTT	Towards an Abstract Lightweight Multi-robot ROS Simulator for Rapid Experimentation
Joshua W. G. GILMOUR	Towards Heterogeneous Modular Robotic Systems for Industrial Applications
James KELLETT	Angled Cantilever Construction by Force-Aware Robotic Swarms

POSTER Session 2.1

Lecture Theatre 1 (LT1)

PRESENTER	POSTER
Rachel TRIMBLE	Skid-steer friction calibration protocol for digital twin creation
Barry William MULVEY	Traversing Visually-Similar Physically-Different Obstacles with DeforMoBot, a Bio-Inspired Deformable Mobile Robot
Laura ALVAREZ-HIDALGO	Human evaluation of robotic grippers for berry picking
Kedar SUTHAR	Casting vs injection moulding: a comparison study for in-lab low-cost soft robot fabrication
Saloni HAJARE	Suction Cup Detachment Mechanism based on Fluidic Soft Actuators Strain for In-vivo Applications
Kavyan ZOUGHALIAN	Access Control Mechanism Framework for Assistive Robots in Healthcare
Elijah ALMANZOR	Comparative study of hand-tracking and traditional control interfaces for remote palpation
Chapa SIRITHUNGE	Estimation of Soft Body Deformation by Using Light
Srikishan VAYAKKATTIL	Plant phenotyping using DLT method: Towards retrieving the delicate features in a dynamic environment
Huijiang WANG	Reduced-Order Modeling of a Soft Anthropomorphic Finger for Piano Keystrokes
Joelle SOGUNRO	Multi-axis Force and Tactile Sensor Sleeves for Micro Catheters & Cannulas
Rafi HOSSAIN	Prototyping Mechanical Design of a Robotic Tail for Human Balance Support
Elijah ALMANZOR	Rapid Development and Performance Evaluation of a Potato Planting Robot

POSTER Session 2.2

Lecture Theatre 2 (LT2)

PRESENTER	POSTER
Andrew WEST	Sonification of Ionising Radiation Data for Robot Operators
Burak KIZILKAYA	5G-based Low-Latency Teleoperation: Two-way Timeout Approach
Nasiru ABOKI	Automating a Telepresence Robot for Human Detection, Tracking, and Following
Hyeonggeun YUN	Generative Model-based Simulation of Driver Behavior when Using Control Input Interface for Teleoperated Driving in Unstructured Canyon Terrains
Seth ROBERTS	Human Machine Interface for a Pick-and-Place Task Using Extended Reality Devices
Aaron SMILES	Implementation and Validation of a Stereo Vision System for Robotic Teleoperation Simulator
Emanuele DE PELLEGRIN	Towards Multimodal Sensing and Interaction for Assistive Autonomous Robots
Imene TARAKLI	Personalised Interactive Reinforcement Learning with Multi-Task Pre-training

Skid-steer friction calibration protocol for digital twin creation

Rachel Trimble[1] and Charles Fox[2]

[1]Department of Plant Sciences, University of Cambridge
[2]School of Computer Science, University of Lincoln

Abstract. Mobile robots require digital twins to test and learn algorithms while minimising the difficulty, expense and risk of physical trials. Most mobile robots use wheels, which are notoriously difficult to simulate accurately due to friction. Physics engines approximate complex tribology using simplified models which can result in unrealistic behaviors such as inability to turn or sliding sideways down small slopes. Methods exist to characterise friction properties of skid steer vehicles [1] but use has been limited because they require expensive measurement equipment or physics models not available in common simulators. We present a new simple protocol to obtain dynamic friction parameters from physical four-wheeled skid-steer robots for use in the Gazebo robot simulator using ODE (Open Dynamics Engine), assuming only that calibrated IMU (Inertial Measurement Unit) and odometry, and vehicle and wheel weights and geometry are available.

1 Applicability of Friction Models in Gazebo

Coulomb friction models the typical observation that dry friction F_C opposes an external pulling force F_e up to a limit proportional (μ) to the normal force F_n at the contact, $F_C = \min(\mu F_n, F_e)$. F_n is the reaction of the supporting surface to the weight of the object. This is often visualised as a cone, such that the object moves (or a wheel loses traction) when the resultant force vector is outside the cone. In ODE this cone may be approximated by a pyramid which is faster to compute. ODE also allows different μ values in longitudinal and latitudinal directions, specified separately as $fdir$. If a vehicle does not need to slip or skid (e.g. it has *two* wheel differential drive and does not travel at high speeds) then a high μ value alone would give reliable movement without slip. The μ values control *maximum* friction forces which can be applied, which are rarely needed in full, so for wheels, where friction is generally desirable, it is common to set them to infinity or 'any high number' > 1. This model is not sufficient in the case of four-wheel skid-steer robots because the wheels do not point in the direction of motion as the robot turns and the wheels simultaneously drive and slip.

Slip friction (FDS) models observed friction in contacts mediated by a lubricating liquid or sand-like particles. Slip friction force is proportional to the

sliding velocity, $F_S = \frac{v_{slip}}{k}$. ODE allows two separate k slip values for perpendicular directions defined by $fdir$. Using these in longitudinal and lateral directions is a potential mechanism for skid steer vehicles to turn and is a valid approach for deformable surfaces such as sand, but is not realistic from vehicles on hard surfaces because it allows a stationary vehicle to slip sideways down infinitesimal slopes. Most robots are best programmed to avoid lubricated surfaces so this model is not our focus.

wheel_slip is a linearized approximation to the brush tyre model [2], which assumes the wheels are made of stretchable tyres approximated as flexible brushes. In this model, the slip is caused by new tyre elements hitting the road and then deforming as the wheel turns. Hence, a non-turning wheel will not slip and a turning wheel will slip with a slip constant is proportional to the *rotation speed* (v_r) of the wheel, as $F_W = \frac{v_{slip}}{cv_r}$. This allows slips for turning while preventing stationary vehicles from slipping down slopes. wheel_slip is not included in core ODE but as a Gazebo plugin written in ODE. The plugin works by *dynamically* updating the ODE slip parameter, as used statically in Slip friction (FDS) above.

2 Proposed Protocol

Contact softness First, tune the number and location of wheel-surface contacts to give one reliable contact per wheel, using ODE parameter kp to govern the contact 'softness'. In ODE, friction is independent of contact area, so contacts can be modelled as single points. If kp is too soft, the robot can get two contacts per wheel or sink into the ground. Contacts should be in the center of the wheel, not towards either edge: real wheels usually have a curved profile which should be simulated; or using a very narrow simulated wheel can be a close approximation.

Coulomb parameters The limit of Coulomb friction is generally not desirable to hit as it represents skidding out of control. This makes it difficult to test safely with larger robots and so an arbitrary large value such as 10 or 1000 is appropriate for many applications. If it is really needed, options include running the robot into a fixed object until the wheels spin and measuring the force applied or measuring the braking distance for an emergency stop. Tables of μ for standard surface type pairs are widely available, ranging from 0.05 (teflon-teflon) to 1.2 (rubber-asphalt). The $fdir$ parameter needs to be set to ensure that the longitudinal and latitudinal parameters follow the orientation of the robot. Pragmatically, this is most easily checked by setting one slip parameter much larger than the other and tilting the world using the ODE gravity parameters.

wheel_slip setup this ODE plugin should be enabled, with normal force 1.0. The normal force is used as an extra multiplier to correct units but is not practically useful over and above the wheel slip constant as there is no capability to dynamically vary the force. The integration of the plugin to the SDF file is shown in the source code [3].

Step size tuning If the step size is too large, the response to large differences in long-lat slip is 'damped'. To tune step size, the lateral slip test below can be set up in simulation and the achieved ratio measured.

Longitudinal wheel_slip Initial odometric calibration should be done at the slowest practicable speed such that the wheelslip experienced by the robot is minimised. Subsequently, to calibrate the longitudinal wheel_slip, the robot is set to move a fixed distance (d_t) at a target velocity (v_r) as determined by its odometry. The acceleration, a, experienced by the robot is measured by the IMU to determine the force profile on the robot and the longitudinal wheelslip parameter (c_{long}) tuned such that the overall slip is consistent with the distance seen by the wheel odometry. The test is proposed as a step function between a lower start speed and a higher target speed such that numerical errors around low velocities are minimised. As the vehicle moves straight forward, acceleration is provided entirely by longitudinal friction of the wheels on the ground,

$$a = \frac{F}{m} = \frac{4(v_r - v)}{c_{long} v_r m}, \quad v_r = \frac{v}{1 - 4ac_{long}m}. \tag{1}$$

Acceleration can be numerically integrated to find velocity $v(t)$ and the c_{long} parameter tuned by search such that

$$d_t = \int \frac{v}{1 - 4ac_{long}m} dt. \tag{2}$$

Lateral wheel_slip This test involves the robot being spun on the spot given a constant target angular velocity ω_{target} and the achieved steady state angular velocity ω measured by counting the time taken to make n rotations. This provides a ratio between the latitudinal and longitudinal slip parameters. Care must be taken during the test that neither latitudinal nor longitudinal slip is in the Coulomb region where μ is dominant. This can be back-calculated by comparing the slip forces against μN after the test and rerunning with a lower ω_t if required. For the tested robot model, the centre of mass was coincident with the centre of the wheelbase but a correction term would be needed if this was offset.

Figure 1 shows the force balance for the test. Resolving moments about the robot centre, $bF_{long} = lF_{lat}$,

$$\Rightarrow F_{long} = \frac{b\omega_{target} - b\omega}{b\omega_{target} c_{long}}, \quad F_{lat} = \frac{l\omega}{b\omega_{target} c_{lat}} \Rightarrow \frac{c_{lat}}{c_{long}} = \frac{l^2\omega}{b^2(\omega_{target} - \omega)}. \tag{3}$$

3 Validation

A Pioneer P3AT simulation with known parameters was used to generate data using the protocol: $c_{long} = 0.005, c_{lat} = 0.05, \omega_{target} = 1.0, CFM = 0, ERP = 0.2, Stepsize = 0.0001$. We then attempted to recover c_{long}, c_{lat} from the data. The cone friction model was used as conceptually more accurate. Both directions' μ were set to 10.

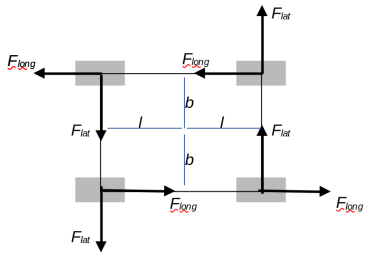


Fig. 1: Force balance for a skid steer robot spinning on the spot.

Rotation test The ratio $\frac{c_{lat}}{c_{long}}$ was calculated from the data by eqn. 3, as 9.998 (0.2% error). The simulation was not exactly aligned due to the finite step size but this result shows that the model and the equations are consistent. For physical experiments, a small range of ω would be tested to ensure the assumed slip region linearity was plausible.

Longitudinal test target velocity of 1.0m/s and test duration 1s were used. Torque limit was reduced to limit vehicle acceleration and prevent 'wheelies', and the instantaneous accelerations and velocities calculated numerically from reported positions. The 'encoder' distance was determined using the wheel angular velocities and the velocity and acceleration signals calculated numerically from 20Hz position measurements. A maximum limit for c_{long} was determined according to $c_{long} < \frac{4}{a_{max}m}$ (to avoid div0 errors) and bisection used to find the matching c_{long}. This test was repeated for a range of c_{long} values but consistently underestimated c_{long} (the contacts looked 'stiffer' than they were supposed to) and plots of acceleration were very noisy. This is thought to be due to the use of numerical differentiation. This effect would not be present in real life tests as the acceleration could be measured directly with an IMU. However, the measurement is fundamentally challenging because the amount of slip demonstrated over the length of a lab floor would be expected to be small. In cases where the measurement is not repeatable, it may be more appropriate to pick a stiff c_{long} based on IMU resolution (i.e. 'based on these quick tests, we know the tyres are at least this grippy').

The validation suggests that the new protocol may work to reduce the time currently spent by modellers performing manual search for realistic parameters.

References

1. S. Khaleghian, A. Emami, and S. Taheri, "A technical survey on tire-road friction estimation," *Friction*, vol. 5, pp. 123–146, 2017.
2. H. B. Pacejka, "Theory of Steady-State Slip Force and Moment Generation," in *Tire and Vehicle Dynamics*. Butterworth-Heinemann, 2012, pp. 87–147.
3. "osrf/gazebo," Jul. 2022, original-date: 2020-04-19T04:26:21Z. [Online]. Available: https://github.com/osrf/gazebo/blob/gazebo11/plugins/WheelSlipPlugin.hh

Traversing Visually-Similar Physically-Different Obstacles with DeforMoBot, a Bio-Inspired Deformable Mobile Robot*

Barry William Mulvey, Thilina Dulantha Lalitharatne, and Thrishantha Nanayakkara

Dyson School of Design Engineering, Imperial College London, London, UK
b.mulvey21@imperial.ac.uk

Abstract. Many animals have evolved to be able to change their natural shape in order to fit through narrow gaps and spaces. This is generally not possible with rigid robots. In this work, DeforMoBot (a deformable mobile robot) is tested attempting to traverse obstacles of various masses (and thus, movabilities). The robot uses a shape-adjustment algorithm so that it can traverse obstacles rather than having to circumnavigate them. Our results demonstrate the inverse relationship of the robot's degree of deformation and its ability to push against obstacles in its path.

Keywords: Biologically-inspired robots · Compliant joints and mechanisms · Deformable robots · Field robots · Whisker-based navigation

1 Introduction

The embodied intelligence of robots is crucial for achieving higher performance and enhanced capability in their environments [1,2,3]. In uncertain surroundings, robots regularly encounter obstacles in their desired or chosen paths. By analysing and investigating the properties of these obstacles, robots can significantly improve their efficiency by adapting their body shape to achieve higher chances of success at traversing such obstacles. The ability to transform their shape could even be critical to mission success.

Animals have evolved to show remarkable adaptability to their habitats and other unstructured settings. Many can fit through gaps smaller than their natural body shapes [4]. Cockroaches' ability to navigate confined spaces inspired an origami-style, soft, legged robot [5]. The caterpillar-inspired soft robot GoQBot can switch between crawling and rolling [6]. A sprawl-tuned autonomous robot (STAR) uses variable leg sprawl to traverse obstacles [7], while the Weaver robot was augmented to adapt its walking posture to navigate in confined spaces [8]. Cats can adapt their flexible skeletons to squeeze through spaces narrower than their resting body dimensions [9], which has inspired our own robot design.

This work examines the ability of a deformable mobile robot to traverse obstacles which are visually similar but physically different. The robot can change its shape using a tuned real-time algorithm based on proprioceptive feedback.

* This work was partly supported by EU Horizon 2020 research and innovation programme under grant agreement 101016970 (Natural Intelligence for Robotic Monitoring of Habitats) and by UK Engineering and Physical Sciences Research Council (EPSRC) DTP scholarship jointly with Analog Devices, Inc.

2 Method

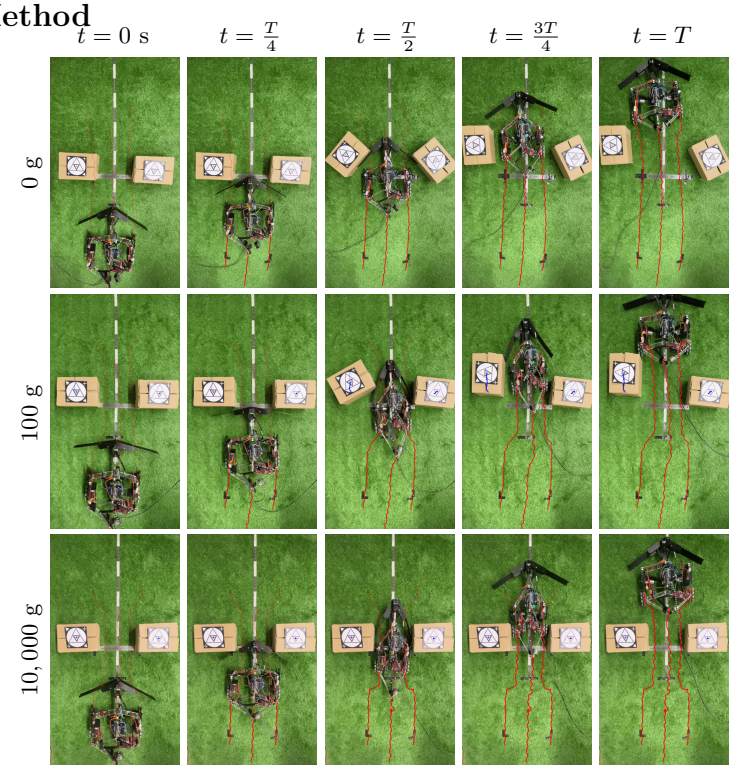


Fig. 1. Comparison of the robot's path (in red) and obstacle displacement (in blue) from the traversal experiments. The masses of the obstacles are shown in the row headers and the timestamps are shown in the column headers, where $T = 5$ s.

We utilise DeforMoBot, a bio-inspired deformable mobile robot [10]. Its shell is composed of acrylic pieces connected together with hinges that allow the robot to change shape from a regular hexagon to an elongated rhombus about 66% of its original width. A linear guide rail acts as the "spine" of the robot, and spring-loaded 3D-printed "whiskers" are attached to its front tip where the orientation of a 6 mm neodymium magnet is measured. These measurement data are sent at a rate of 200 ms to an Arduino Uno which implements an algorithm to determine the robot's optimal body shape. A digital servo connected to a 2-bar linkage controls the shape of the robot. The robot navigates using motor-driven 3D-printed rimless front wheels and passive omni-directional back wheels. Further design specifications and details of our robot can be found in [10].

DeforMoBot aims to traverse obstacles in its path while employing a wide body shape for stability. To achieve these objectives we implement an algorithm, which the robot employs to progress through obstacles efficiently, modelled as

$$\Delta\Phi = a\Delta\Omega^2 + b\Delta\Omega + c \tag{1}$$

where a, b, and c are tuned coefficients coupling the whisker angle Ω with the servo angle Φ and desired body shape. The shape-adjustment algorithm accounts for both the real-time whisker angle deformation and the current body shape. Our setup is controllable; different algorithms can easily be deployed and tested.

Table 1. Summary of Experiment Results

Obstacle Mass [g]	Success Rate	μ_s [cm]	δ_s [cm]
0	100 %	92.50	3.15
100	100 %	93.65	2.48
10,000	95 %	79.45	8.36

μ_s = mean distance travelled by the robot in 5 seconds
δ_s = standard deviation of the mean distance, μ_s

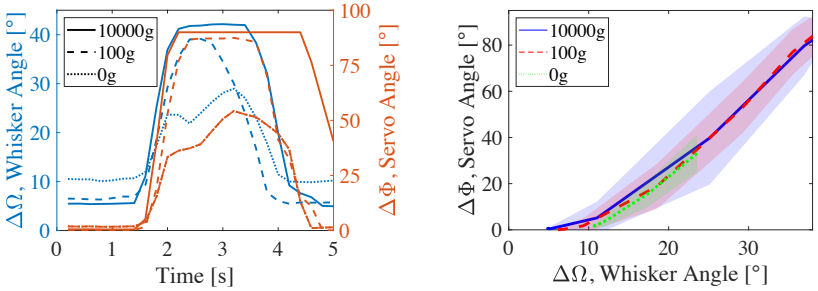


Fig. 2. Left: Mean results of the servo angle reacting to the whisker angle during interaction with obstacles in the robot's path. Right: Comparison of angle changes with each obstacle, where the mean is plotted surrounded by standard deviation shading.

3 Experiments and Results

We performed experiments to observe how the robot interacts with obstacles that have the same visual features but different masses. Two boxes (22 cm × 16.5 cm × 11.5 cm) were placed 24 cm apart (the narrowest robot width) 20 cm in front of the robot. Various masses were placed inside these boxes, detailed in Table 1. Based on the obstacle masses (and thus movabilities), the robot could push these away, narrow its body to squeeze through, or combine these actions.

DeforMoBot completed 20 trials attempting to traverse each obstacle. The experiment results are summarized in Table 1, where μ_s is the mean distance travelled by the robot in 5 seconds and δ_s is the standard deviation. The robot successfully traverses the lighter obstacles with little distance deviation. The heavier obstacles are more challenging, with greater distance deviation.

Fig. 1 shows examples of experimental results. Visual markers track the movements of the robot and the obstacles. When the robot can push away lighter obstacles, it only slightly adjusts its body shape. Conversely, when it cannot move the obstacles, it narrows its shape until it has successfully traversed the obstacles at which time it can resume its natural body shape.

Fig. 2 compares how the whisker and servo angles change during interaction with the obstacles. The shape adjustment is in real-time, apart from a widening delay which occurs only after the robot has traversed the obstacles. The left plot shows steeper gradients when heavier obstacles are encountered. Direct comparisons between the measured whisker angles and resultant servo angles are plotted on the right. The robot only slightly deforms upon encountering the lightest obstacles. Heavier obstacles result in greater deformation and higher entropy which is represented by the greater variability in their standard deviations.

These results show the robot's degree of deformation is inverse to its ability to push against obstacles in its path. These capabilities can facilitate improved perception and proficiency of robots and allow them to navigate efficiently and effectively, particularly in challenging and unstructured environments. Future work will involve further design development and progression of our methods.

4 Conclusions

This work examines the deformable mobile robot DeforMoBot traversing visually similar obstacles of various masses. By adapting its body shape when encountering obstacles, the robot can successfully traverse obstacles rather than having to circumnavigate them. It achieves this using a real-time shape adjustment algorithm which accounts for the robot's current shape and proprioceptive whisker feedback. The robot was assessed traversing obstacles placed apart at a distance smaller than the robot's normal size. It achieved 100% success rate traversing lighter movable obstacles (slightly adjusting its body shape), 100% success rate traversing more moderate obstacles (using a combination of shape adjustment and pushing the obstacles), and 95% success rate traversing heavy obstacles which cannot be moved (relying fully on its shape-changing ability). Our results demonstrate the inverse relationship of the robot's degree of deformation and its ability to push against compliant obstacles in its path.

References

1. R. Pfeifer and J. Bongard, *How the body shapes the way we think: a new view of intelligence*. MIT Press, 2006.
2. S. Kim, C. Laschi, and B. Trimmer, "Soft robotics: a bioinspired evolution in robotics," *Trends in Biotechnology*, vol. 31, no. 5, pp. 287–294, 2013.
3. D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.
4. M. H. Dickinson, C. T. Farley, R. J. Full, M. Koehl, R. Kram, and S. Lehman, "How animals move: an integrative view," *Science*, vol. 288, no. 5463, pp. 100–106, 2000.
5. K. Jayaram and R. J. Full, "Cockroaches traverse crevices, crawl rapidly in confined spaces, and inspire a soft, legged robot," *Proceedings of the National Academy of Sciences*, vol. 113, no. 8, pp. E950–E957, 2016.
6. H.-T. Lin, G. G. Leisk, and B. Trimmer, "GoQBot: a caterpillar-inspired soft-bodied rolling robot," *Bioinspiration & Biomimetics*, vol. 6, no. 2, p. 026007, 2011.
7. D. Zarrouk, A. Pullin, N. Kohut, and R. S. Fearing, "STAR, a sprawl tuned autonomous robot," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 20–25.
8. R. Buchanan, T. Bandyopadhyay, M. Bjelonic, L. Wellhausen, M. Hutter, and N. Kottege, "Walking posture adaptation for legged robot navigation in confined spaces," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2148–2155, 2019.
9. F. Richmond and D. Bakker, "Anatomical organization and sensory receptor content of soft tissues surrounding upper cervical vertebrae in the cat." *Journal of Neurophysiology*, vol. 48, no. 1, pp. 49–61, 1982.
10. B. W. Mulvey, T. D. Lalitharatne, and T. Nanayakkara, "DeforMoBot: A bio-inspired deformable mobile robot for navigation among obstacles," *IEEE Robotics and Automation Letters*, vol. 8, no. 6, pp. 3828–3835, 2023.

Evaluation of OSMC open source motor driver for reproducible robotics research

Elijah Alabi[1], Fanta Camara[1,2] and Charles Fox[1]

[1] School of Computer Science, University of Lincoln
[2] Institute for Safe Autonomy, University of York.

Abstract. There is a growing need for open source hardware (OSH) subcomponents to be evaluated. Most robotic systems are ultimately based upon motors which are driven to move either to certain positions, as in robot arms, or to certain velocities, as in wheeled mobile robots. We evaluate a state of the art OSH driver, OSMC, for such systems, and contribute new Open Source Software to control it. Our findings suggest that OSMC is now mature enough to replace closed-source motor drivers in medium-size robots such as agri-robots and last mile delivery vehicles.

1 Introduction

Robotics research currently struggles with reproducibility. While open source software (OSS) such as ROS enables software reuse, this is not so for hardware, which is proprietary and differs between labs. Researchers thus waste significant time porting software to run on their different hardware platforms. Researchers in developing counties often cannot afford the proprietary robots used by other labs. Open source hardware (OSH) is hardware whose designs and build instructions are made public, easy, and low-cost so that anyone may build and modify, enabling large community collaborations to grow. Fully open software and hardware stacks would allow any researcher to download, build, exactly replicate, then extend the published work which they read.

Recent 'shallow' definitions of OSH such as the 2020 CERN-OSH licences [4] do not require designs to be made up of OSH subcomponents, but allow closed source subcomponents if available on the open market. 'Deep' definitions such by Open Source Ecology (OSE) [7] further restrict subcomponents, recursively, to be all OSH, so that entire designs are open down to the level of ISO standard nuts, bolts, resistors and transistors. 'Deeper OSH' has been proposed [6] as the process of successively replacing lower level subcomponents of shallow OSH designs with OSH alternatives, working towards deep OSH. To enable this process, there is a need for standard subcomponents to be not only created as OSH but also to be publicly evaluated to scientific standards. A recurring difficulty is that while some OSH designs are created and published through academic peer-review, others emerge from the maker community outside the academic system. Maker designs may be of high quality but additional peer-reviewed evaluations (sometimes known as 'blessings') are then needed to create specifications and/or create trust that the specifications are met.

Motor drivers are a common subcomponent so an important target for deeper OSH. We here perform the evaluation of a maker community OSH motor driver, the OSMC (Open Source Motor Controller; *sic.*—it is a driver not a controller), for this purpose. OSMC is a brushed DC driver designed by hobbyists in 2001 for use in *Robot Wars* style, manually remote controlled combat robots. OSMC as been used [8] for a telecontrolled service weeding robot in outdoor environment; in an underwater vehicle [1]; in the RoboCup Rescue challenge [2]; and in the manual control [9] "Battle-Bot" combat robot. However these projects are not open source hardware and do not provide evaluations of the OSMC other than reporting its use in the larger systems. OSMC has been released as a hardware design but does not include standard software to control it. This paper aims to address these gaps by evaluating OSMC, and by releasing new open software to control it, available at `https://github.com/Pelex-a/osh-motor-control`.

2 Methods

We evaluate OSMC for use in both 'small-size' (capabale of motion and information gathering) robots such as RC cars, and 'medium sized' (capable of actuating physical work) robots such as agri-robots and last-mile delivery vehicles [3].

The small-size evaluation setup is made up of the L298N driver module, a DC motor, an incremental encoder and a means of serial port communication, with new Arduino software for PID control as shown in fig. 1.

The medium-size evaulation uses OSMC with two example motors. The motors used were a 73ZYT-155-24-MCP4-25:1 rotary DC motor (Motion Control Product Ltd) shown in Fig. 2 and a GLA750-P 12V DC linear actuator with potentiometer feedback shown in Fig 3. The two main wires of the motor are connected to the 'MOT+' and 'MOT-' leads on the OSMC driver. For directional and PWM control, AHI, BHI, ALI, BLI and Disable pins are connected to digital pins of an Arduino board. The OSMC is then connected to the DC power source. An external incremental encoder is connected to the Arduino. The linear actuator has the same connection from the OSMC to the microcontroller board. However for control, we connected the potentiometer wiper, which is one of the three leads of the potentiometer in our linear actuator, to the analogue pin of the Arduino board. The two other leads, the potentiometer reference, are connected to the ground and reference voltage pin on the Arduino board.

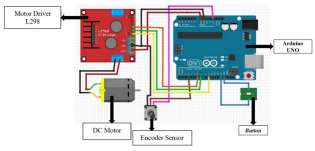


Fig. 1: Hardware wiring

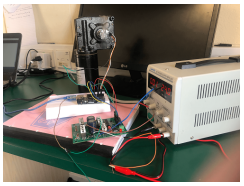


Fig. 2: Rotary setup

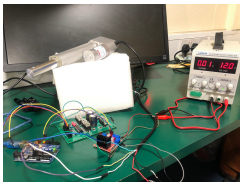


Fig. 3: Linear setup

3 Results

An open-loop test was conducted to observe the system response of the small-scale set-up by monitoring the revolution made per minute with a constant input voltage in PWM. The system rises initially from rest and settles at a specific RPM over time. It is noticed that the open loop system does not react to disturbance or change in condition. For this reason, a system with feedback – a closed system – is needed to efficiently carry out the desired task and to ensure the targeted value is reached.

Proportional-Integral-Derivative (PID) control is widely used in closed loop systems. We test PID control to evaluate system performance responding to angular position and speed requests. Videos of the tests are at `https://bit.ly/3qrC4JL` and `https://bit.ly/3waN9zG`. They show high degrees of positional and speed accuracy ($Accuracy(\%) = |\frac{set\ value - observed\ position}{set\ value}|\ X\ 100$) as seen in Tables 1 and 2. Each of the three control techniques has varying effect on the system as shown in Table 3, 4 and 5.

Table 1: Rotary performance

Set Value (mm)	Motor Position (mm)	Accuracy (%)
90°	91°	98.89%
180°	182°	98.89%
270°	272°	99.23%
360°	363°	99.17%

Table 2: Linear performance

Set Value (mm)	Actuator Pos (mm)	Accuracy (%)
10	7	70
25	23	92
40	39	97.5
70	68	97.1
85	83	97.6

Increasing the proportional (P) action in the control system reduces the steady-state error. Adjusting the integral (I) constant effect changes by eliminating the residual steady-state error. Although, the system becomes unstable and vibrate aggressively when set too high. Lastly, the derivative (D) action controls overshooting and rise time. These recordings (`https://bit.ly/3EN65HP` & `https://bit.ly/3wvRv4L`) shows how the system performed.

4 Discussion

The evaluation suggests that OSMC is ready for practical use in medium-size robots such as last-mile delivery vehicles and agri-robots. We have released new open source Arduino control software for OSMC, which now makes OSMC plug-and-play. This enables other community members to focus next on creating, building, and releasing new OSH robots using OSMC as a subcomponent. For example, OSMC is now being integrated into a general purpose medium sized robot control board [5]. Future work could extend OSMC with regenerative braking,

Table 3: System Response, P

Kp	**Ki**	**Kd**	Rise Time	Steady-state error
2	0	0	-	-
3	0	0	-	-
5	0	0	5.2	26
8	0	0	4.1	19
10	0	0	3.7	12

Table 4: System Response, I

Kp	**Ki**	**Kd**	Rise Time	Over-shoot	Steady-state error
10	0	0	3.7	-	12
10	2	0	3.4	-	0
10	3	0	3.2	4	0
10	4	0	2.9	11	3
10	5	0	2.8	19	5

Table 5: System Response, D

Kp	**Ki**	**Kd**	Rise Time	Over-shoot	Steady-state error
10	4	0	2.9	11	3
10	4	2	2.85	7	1
10	4	3	2.81	5	0
10	4	4	2.77	5	3
10	4	5	2.75	3	2

as found in many closed drivers. OSMC requires some closed subcomponents – including the Intersil H-bridge driver chip – which could be replaced with deeper OSH alternatives. OSMC is a brushed driver, but maker OSH Brushless drivers such as ODrive[1] v3.5 (but not later versions) and OpenBLDC[2] are similar in power to OSMC and could be similarly evaulated.

References

1. Aldehayyat, Y., Dahan, R., Fayyad, I., Martin, J., Perkins, M., Sharples, R.: Ocra-xi: An autonomous underwater vehicle. OCRA AUV Team (2009)
2. Angeles, D., Arroyo, G., Culebro, J., Espinoza, J.L., Gutierrez, H., Lara, M., Minami, Y., Muñoz, S., Ramos, G.: Robocup rescue 2014, team UNAM.
3. Camara, F., Waltham, C., Churchill, G., Fox, C.: OpenPodcar: an open source vehicle for self-driving car research. Jounral of Open Hardware (in press, preprint arXiv:220504454)
4. CERN: CERN Open Hardware Licence https://cern-ohl.web.cern.ch/
5. Gaikwad, K., Soni, R., Fox, C., Waltham, C.: Open source hardware robotics interfacing board. In: TAROS (2023)
6. Henry, A., Fox, C.: Open source hardware automated guitar player. ICMC (2021)
7. Jakubowski, M.: Open Source Ecology. www.opensourceecology.org/ (2003)
8. Khan, M.I.: A framework for telecontrolled service robots. Master's thesis (2008)
9. Warren, J.D., Adams, J., Molle, H.: The Battle-Bot, pp. 513–562. Apress (2011)

[1] www.github.com/madcowswe/ODriveHardware/blob/master/v3/v3.5docs/schematic_v3.5.pdf

[2] www.open-bldc.org/wiki/Open-BLDC

Cable-Loop Gripper Add-on for Multimodal Grasping of Lab Supplies

Lupo Manes[1,2], Sebastiano Fichera[1,2], Hatem Fakhruldeen[1], Andrew I. Cooper[1], and Paolo Paoletti[1,2]

[1] Leverhulme Research Centre for Functional Materials Discovery, Material Innovation Factory, University of Liverpool, Liverpool L69 7ZD
LManes@liverpool.ac.uk Paolo.Paoletti@liverpool.ac.uk
Sebastiano.Fichera@liverpool.ac.uk
[2] School of Engineering, University of Liverpool, Liverpool L69 3GH
Hatem.Fakhruldeen@liverpool.ac.uk aicooper@liverpool.ac.uk

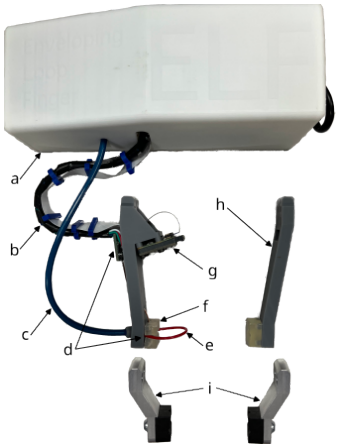
Abstract. This paper proposes a cable-based gripping device that can be integrated into an existing parallel jaw gripper to extend its capabilities. The two devices combined provide complimentary grasping modes, respectively, for grasping small and delicate objects with little clearance required and strong and reliable grasp of large objects. This device can be used to simplify the deployment of robotic manipulators in Chemistry research labs thanks to the flexibility it provides. The cable loop gripper can grasp vials from 10 to 50 mm in diameter and when it is not needed, the parallel jaw gripper can be operated normally. The device we present has been tested by manipulating a variety of lab supplies.

Keywords: Robotic Gripper· Compliant Mechanism· Chemistry Automation

1 Introduction

Automation techniques can be found in the field of chemistry going back over a century [6]. Modern automation relies on computing and complex electromechanical devices to carry out any task a human could, but more efficiently and reliably. We are now at a point where the technology has matured enough that, under some conditions, completely autonomous system can be used to carry out chemistry experiments [3,5,**?**,4,1]. These have become more common in research environments where the higher throughput of an automated system is pivotal to complete experiments in a timely fashion. In such systems, specialised machinery carries out tasks directly related to chemistry including but not limited to measuring, dosing, dispensing both liquids and solids as well as mixing and many of the analytical steps. All of these systems are stand-alone cells that accomplish their task to the standard for accuracy, precision and repeatability required by experiments, forming a vertically-integrated, reliable system. Therefore, a device capable of pick and place operation is a further step forward in the effort to completely automate chemistry experiments. While chemistry automation is very effective, it comes at the cost of great time and resource investment

to set up due to their complexity [2]. Robotic manipulators can alleviate this issue and have been used successfully for chemistry research [8], either as systems that can move throughout an entire lab or inside a designated enclosure. They can be employed to run the equipment and transport chemical-filled vessels to the appropriate locations, freeing up the scientists' time to focus on the intellectual aspects of the experiment [9]. Commercial solutions, including the UniteLab cobotic lab assistant and the Mettler-Toledo Quantos-Chronect robotic stations for solid and liquid dispensing, are respectively available for enclosed and free-roaming methods. Commercially available robotic arms are very adaptable and offer a wide range of features, but the end-effectors that are mounted on them frequently limits the system's potential applications. The Parallel Jaw Gripper (PJG), are still the most popular design for end-effectors [7]. They can yield near perfect results in controlled environments, with bespoke fingers and associated tips designed for the task, but this approach lacks the flexibility to be adopted for different chemistry workflows because of the resources required to set the system up properly.



(a) Control Unit
(b) Electrical connections
(c) Bowden Cable
(d) Singletact Capacitive PDMS force sensor
(e) TPU cable loop
(f) Soft fingertip
(g) Vision Unit
(h) Passive finger
(i) Original Franka Hand fingers

Fig. 1: Control unit, Loop-O finger and Franka Hand Fingers.

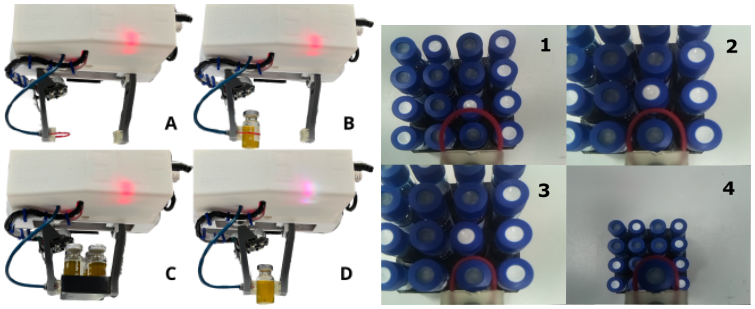
2 Proposed Device

The device presented in this paper, shown in Fig.1, aims to be a plug-in upgrade for PJGs in chemistry labs. The device consists of 3 main components: the control unit, a finger with an embedded Cable Loop Gripper (CLG) and a second passive finger to match the geometry of the CLG. The control unit can be mounted to the main body of the PJG and the finger can easily replace the ones it comes equipped with from the factory, almost all PJG allow the fingers to be swapped. Our implementation was designed to integrated with the Franka Emika Hand,

the PJG provided with the Panda robotic manipulator, but the design can be easily modified to account for different PJGs. The CLG works by sliding a close loop of wire over the target object and then constricting the object against a grasping appendage. Thanks to the working principle, this type of gripper is inherently compliant and requires minimal control effort to successfully grasp delicate objects. The specific implementation proposed here has been designed to handle vials. It has a payload of up to 150 grams and can grip vials between 10 and 50 mm diameter with as little as 3 mmmm clearance between vials stored in a rack. The cable is Ninjatek TPU, a 3D printing material used to manufacture flexible objects. This material has been chosen because it is widely available, extremely durable while remaining flexible enough to maintain a 5 mm bending radius without plastic deformation and stiff enough to hold its own weight without significant sagging while forming loops in our application's required range. A capacitive PDMS force sensor (d) is placed between the rubber like (f) finger tip and the main body of the finger to provide feedback on the grasping force as well as to act as a homing switch. The device includes a vision module (g) consisting of a camera, time-of-flight range-finder and LEDs are placed near the base of the finger pointing directly down at the area where the cable loop can be deployed to provide feedback on the size of the cable, the vials to be grasped, as well as their relative position to adjust for positional errors during grasping. When the CLG is not required, the cable can be fully retracted out of the way and the PJG can operate normally. The PJG can be used to grasp vials too big for the CLG and other objects with geometries that would not work for the CLG, like the vials' storage trays used for ease of transportation.

3 Preliminary Testing and Conclusions

The proposed device has been tested in various pick and place tasks involving real lab supplies. The CLG has been shown to be able to grasp and manipulate all the labware common in chemistry laboratories that would have proved too difficult to grasp with traditional PJG due to their small size or tight clearances around them when stored in trays. The CLG has been tested for reliability by carrying out 1000 thousand pick a place tasks with vials of 3 different diameters (15mm, 22mm, 28mm) and only 8 failures had been recorded. The PJG's performance was unaltered for the purpose of our application throughout testing as the geometry of the fingers is unchanged, and the cable loop can fully retract out of the way. One of the main limitation for the current design is that, when working with small vials, it can only grasp the ones on the edge of the tray, but this could be resolved by reducing the footprint of the finger to allow for access in between the vials. The proposed device effectively combines both the CLG and PJG, thus eliminating the limitations of using only PJGs in chemistry workflows.



(a) Gripper device in different states.

(b) Grasping sequence over a 2ml vial, as seen from the vision module.

Fig. 2: Images from tests.

References

1. Burger, B., Maffettone, P.M., Gusev, V.V., Aitchison, C.M., Bai, Y., Wang, X., Li, X., Alston, B.M., Li, B., Clowes, R., Rankin, N., Harris, B., Sprick, R.S., Cooper, A.I.: A mobile robotic chemist. Nature **583**(7815), 237–241 (2020). https://doi.org/10.1038/s41586-020-2442-2, http://dx.doi.org/10.1038/s41586-020-2442-2
2. Christensen, M., Yunker, L.P.E., Shiri, P., Zepel, T., Prieto, P.L., Grunert, S., Bork, F., Hein, J.E.: Automation isn't automatic. Chemical Science **12**(47), 15473–15490 (2021). https://doi.org/10.1039/d1sc04588a
3. King, R.D., Rowland, J., Aubrey, W., Liakata, M., Markham, M., Soldatova, L.N., Whelan, K.E., Clare, A., Young, M., Sparkes, A., Oliver, S.G., Pir, P.: The robot scientist adam. Computer **42**(8), 46–54 (2009). https://doi.org/10.1109/MC.2009.270
4. Liu, H., Stoll, N., Junginger, S., Thurow, K.: Mobile robot for life science automation. International Journal of Advanced Robotic Systems **10**, 1–14 (2013). https://doi.org/10.5772/56670
5. Manley, J.D., Smith, T.J., Holden, J., Edwards, R., Liptrot, G.: Modular Approaches to Automation System Design Using Industrial Robots. JALA - Journal of the Association for Laboratory Automation **13**(1), 13–23 (2008). https://doi.org/10.1016/j.jala.2007.09.003
6. Olsen, K.: The first 110 years of laboratory automation: Technologies, applications, and the creative scientist. Journal of Laboratory Automation **17**(6), 469–480 (2012). https://doi.org/10.1177/2211068212455631
7. Piazza, C., Grioli, G., Catalano, M., Bicchi, A.: A Century of Robotic Hands. Annual Review of Control, Robotics, and Autonomous Systems **2**(1), 1–32 (2019). https://doi.org/10.1146/annurev-control-060117-105003
8. Sanderson, K.: Automation: Chemistry shoots for the Moon. Nature **568**(7753), 577–579 (2019). https://doi.org/10.1038/d41586-019-01246-y
9. Tabor, D.P., Roch, L.M., Saikin, S.K., Kreisbeck, C., Sheberla, D., Montoya, J.H., Dwaraknath, S., Aykol, M., Ortiz, C., Tribukait, H., Amador-Bedolla, C., Brabec, C.J., Maruyama, B., Persson, K.A., Aspuru-Guzik, A.: Accelerating the discovery of materials for clean energy in the era of smart automation. Nature Reviews Materials **3**(5), 5–20 (2018). https://doi.org/10.1038/s41578-018-0005-z

Towards Long-horizon Motion Planning in Dynamic Environments

Xiaoliang Zhang[0009−0008−9111−1828], Alan G. Millard[0000−0002−4424−5953], and Pengcheng Liu[0000−0003−0677−4421]

Department of Computer Science, University of York, York, UK
{xiaoliang.zhang, alan.millard, pengcheng.liu}@york.ac.uk

Abstract. In contrast to mobile robots, the planning space of manipulators (robot arms) is high-dimensional, which poses great challenges for motion planning. Classical motion planning approaches face computation difficulty in such scenarios, and dynamic obstacles exacerbate the problem. We propose a Deep Reinforcement Learning (DRL) based motion planning approach, with a Gaussian Mixture Regression (GMR) sampling tool, to tackle this problem. Benefiting from the reduction of training difficulty in DRL networks, the proposed architecture is effective for a long-horizon manipulation tasks in dynamic environments.

Keywords: Motion planning · Manipulators · Deep Reinforcement Learning · Gaussian Mixture Regression · Dynamic environments.

1 Introduction

Motion planning (MP) for manipulators with dynamic constraints is a challenging problem, but essential for safe human-robot collaboration (sHRC), where robots must avoid human bodies (dynamic constraints) in real-time. Sampling-based motion planners (SBMP) are often used for MP in high-dimensional space. However, SBMP still faces limitations in very high-dimensional space [1], and faces difficulty in processing dynamic environments [2] due to its high computational cost. On the other hand, Deep Reinforcement Learning (DRL) has been widely used in MP, and model-free DRL is especially powerful in unknown environments. Unfortuantely, DRL methods face difficulty in long-horizon manipulation tasks due to the extremely large search space to explore [3], especially when there is a sparse reward. Gaussian Mixture Models (GMM) are simple to implement, and take uncertainty in the environment into consideration, but they are not good at avoiding geometric constraints [4].

In this paper, we propose a hybrid MP model for manipulators in dynamic environments with GMM/GMR (Gaussian Mixture Regression) and DRL. Concretely, a GMM/GMR is trained from demonstrations of an expert in a certain task, then it is used as an aiding tool to train a DRL model. In the training of DRL model, the GMM/GMR is applied to bias the training of DRL, thus reducing its searching space to help it converge.

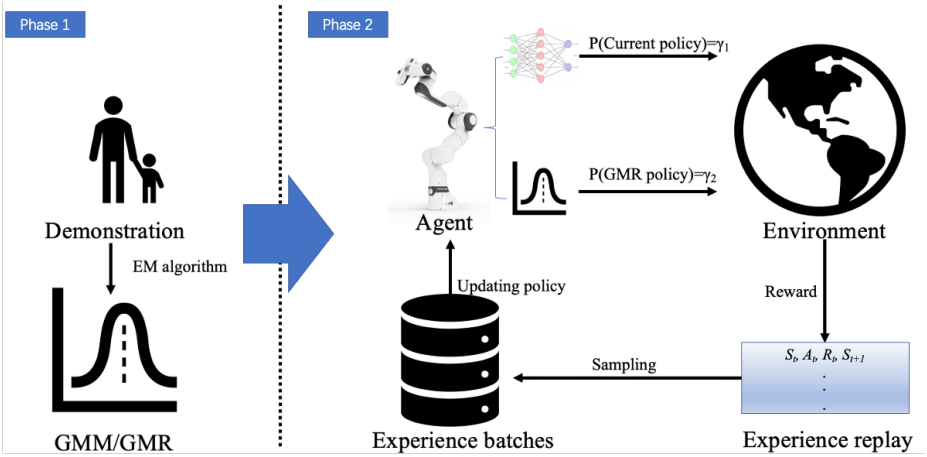


Fig. 1. Architecture of the Proposed Model

The advantages of this work include: (1) by the help of a trained DRL, it can respond to a dynamic environment in real-time. (2) GMR can help train the DRL, by reducing the search space and decreasing its training difficulties, thus for long-horizon tasks that pose challenges in the training of DRL, our proposed method is more powerful. (3) it can process the MP in high-dimensional space, as DRL is well-suited to high-dimensional space.

2 Methodology

We propose a hybrid model for a long-horizon manipulator MP with dynamic constraints, consisting of DRL and GMR. The GMM/GMR, trained by demonstration from humans, functions as a exploration guiding tool for the following DRL-based MP model. The DRL-based MP model can generate motions at the next time step based on the current state, and goal configuration. Instead of exploring the space just by its own policy, the exploration is biased towards imitating the behavior of the GMM/GMR, the search space is thus reduced. The architecture of our proposed model is shown in Figure 1.

2.1 The GMM/GMR Exploration Guiding Tool

We develop a GMM/GMR model similar to [4], which maps time to state. To train the GMM/GMR, we first plan to collect N demonstrations from human experts, each of which will be abstracted into T-time steps. GMM/GMR is the mixture of several Gaussian models (Gaussian components), the equation is as shown in [5] , for example, we have K Gaussian components, so:

$$P(x \mid \theta) = \sum_{k=1}^{K} \alpha_k \phi_k(x \mid \theta_k) \tag{1}$$

where θ_k refers to the mean and covariance matrix of the k-th Gaussian component, i.e., (μ_k, Σ_k). And $\phi_k(x \mid \theta_k)$ is the k-th Gaussian component base. $x = (x_i), i = 1, 2...N$ is the states. By introducing time t, we have:

$$\begin{aligned} P(x \mid t, k) &\sim \mathcal{N}(x; \hat{x_k}, \hat{\Sigma_k}), \\ \hat{x_k} &= \mu_{x,k} + \Sigma_{xt,k}(\Sigma_{tt,k})^{-1}(t - \mu_{t,k}), \\ \hat{\Sigma_{xx,k}} &= \Sigma_{xx,k} - \Sigma_{xt,k}(\Sigma_{tt,k})^{-1}\Sigma_{tx,k} \end{aligned} \tag{2}$$

Through an Expectation-Maximisation (E-M) algorithm, we can train the GMR model and finally the GMR-based exploration tool. This will be implemented with the DRL-based motion planner introduced in the next subsection.

2.2 GMR-guided Soft Actor-Critic

After we get $P(x|t)$, we begin to design the DRL-based planner. We plan to apply the Soft Actor-Critic (SAC) model to train our DRL network, due to its strong adaptability to continuous space and outstanding capability in convergence. To reduce the search space, we make an improvement in the exploration process of the SAC. Typically, the SAC agent will execute actions based on its current policy to collect transitions from environment. Instead, we want the agent to perform exploration with probability γ_1, or follow GMR with probability γ_2 for the whole episode exploration at the beginning of each one, which could reduce the search space. We make γ_1 and γ_2 dynamic, as we want the agent to follow the GMR more often in the beginning, since at this time the GMR model is more experienced. Whereas with the improvement of the agent's policy, the weight explores on its own policy more. This means that γ_1 will increase with training and γ_2 will decrease.

We name our SAC-based planner with GMR as GMR-guided SAC (GSAC), and after it is trained offline, it can be set up for online usage, thus saving computation time when used, compared with SBMP. To make it compliant with dynamic obstacle avoidance, we need to design its reward function carefully like in [6]. Anther concern is the inefficiency brought by sparse reward in a high-dimensional space. We plan to use a composition of dense rewards to provide the agent with timely feedback in case it gets lost in the large space. A typical approach is to give the agent a small value in every step, to query whether it is in collision or reach the goal. And once it happens that the agent collides with obstacles or reaches the goal, a large reward can be given as a inspiration or a punishment. We will design our reward function based on the above considerations, and make improvement to make our model better. The architecture of a SAC is rather complicated, which consists of 4 networks, one policy network, one state value network, and two Q networks. We will implement these four networks and their objective functions in detail in the next steps.

We will conduct the experiments on a Franka Panda robotic arm, a robot with 7 DOFs. The action is represented by the deviation of each joint, $a_t = \Delta q_t$.

There should be an upper limit set on this deviation to avoid dealing damage to the robot. To make the agent collect enough information about the environment, we make state $s_t = (q_t, q_e, d_{obs}, d_{tar})$, where q_t represents the coordinate of each joint, q_e is the position of the end-effector, d_{obs} is the distance between each link of the robot to the obstacles and d_{tar} is the distance of the end-effector to the target. However, the state and action space is still under evaluation and will be further improved in the future based on the experiment results.

Our proposed work is built on the backbone of DRL-based MP in dynamic environments, which is already verified in many past works like in [6] [7]. Based on these works, we introduce a sampling method to enhance its performance, thus it is plausible that our proposed model will perform well in the experiment.

3 Conclusions

In this paper, we introduced a DRL-based MP model to solve a long-horizon manipulation task in dynamic environments, with the help of an innovative GMM/GMR-based exploration strategy. Previous works using DRL-based model for manipulation in dynamic environments demonstrated effectiveness and thus we can conclude that our work will be more effective. We will move on to improve our design and implement the experiment. Moreover, this model also has the potential to be integrated with the ISO requirements for the sHRC.

References

1. Ichter, B., Pavone, M.: Robot motion planning in learned latent spaces. IEEE Robotics and Automation Letters **4**(3), 2407–2414 (2019)
2. Yamada, J., Lee, Y., Salhotra, G., Pertsch, K., Pflueger, M., Sukhatme, G., Lim, J., Englert, P.: Motion planner augmented reinforcement learning for robot manipulation in obstructed environments. In: Conference on Robot Learning. pp. 589–603. PMLR (2021)
3. Nasiriany, S., Liu, H., Zhu, Y.: Augmenting reinforcement learning with behavior primitives for diverse manipulation tasks. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 7477–7484. IEEE (2022)
4. Wang, J., Li, T., Li, B., Meng, M.Q.H.: Gmr-rrt*: Sampling-based path planning using gaussian mixture regression. IEEE Transactions on Intelligent Vehicles **7**(3), 690–700 (2022)
5. Cheng, Q., Zhang, W., Liu, H., Zhang, Y., Hao, L.: Research on the path planning algorithm of a manipulator based on gmm/gmr-mprm. Applied Sciences **11**(16), 7599 (2021)
6. Chen, P., Pei, J., Lu, W., Li, M.: A deep reinforcement learning based method for real-time path planning and dynamic obstacle avoidance. Neurocomputing **497**, 64–75 (2022)
7. Wang, Y., Kasaei, H.: Obstacle avoidance for robotic manipulator in joint space via improved proximal policy optimization. arXiv preprint arXiv:2210.00803 (2022)

Open source hardware robotics interfacing board

Kshitij Gaikwad, Rakshit Soni, Charles Fox, Chris Waltham

[1] School of Computer Science, University of Lincoln

Abstract. Robotics research still struggles with reproducibility. The ROS ecosystem enables reuse of software, but not hardware. Researchers waste time porting systems between hardware platforms to reproduce research between labs. Researchers in developing counties in particular often cannot afford the proprietary robots used by others. If a published robotics system is dependent on any component that is only available from a single supplier, then all work building on it is at risk if that supplier vanishes, de-lists or changes the product. Open Source Hardware (OSH, [3]) is hardware whose designs and build instructions are public, easy, and low-cost so that anyone is free to build and modify them, enabling large community collaborations. Combined open software and hardware stacks allow any researcher to download, build, exactly replicate, then extend the published work which they read about.
A key component of any robot is the interface between ROS and motors. New robots often use arbitrary, messy mixtures of closed and open motor drivers and error-prone physical mountings, wiring, and connectors to interface them. There is a need for a standardizing OSH component to abstract this complexity, as Arduino did for interfacing to smaller components. We present a OSH printed circuit board to solve this problem once and for all. On the high-level side, it interfaces to Arduino Giga – acting as an unusually large and robust shield – and thus to existing open source ROS software stacks. On the lower-level side, it interfaces to existing emerging standard open hardware including OSH motor drivers and relays, which can already be used to drive fully open hardware wheeled and arm robots. This enables the creation of a family of standardized, fully open hardware, fully reproducible, research platforms.

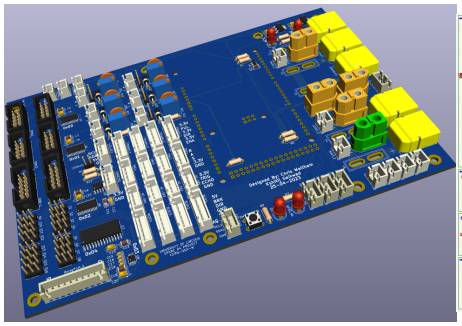
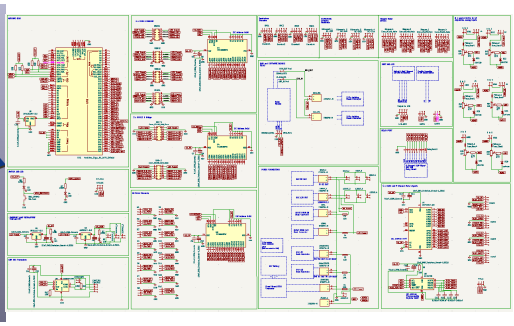
1 Use cases

The board is designed for use in 'medium sized' robots, which we define as robots capable of usefully transporting materials and goods but not people. Three specific demonstrator systems are used in development: *OpenPodCar* [2] is a mobility scooter converted for autonomous driving. It has 24V batteries, is steered using a linear actuator, and uses lidar for sensing and a Raspberry Pi for ROS. Rear wheels are driven by a closed source driver from the donor vehicle or this may be replaced with the open OSMC driver [1]. It has the highest power requirement, drawing up to 10A-15A. *OpenScout* (`https://github.com/cbedio/OpenScout`) is a small 4-wheeled differential drive information gathering robot. The PCB can control the wheels individually using two DHB-12 dual

motor drivers connected to the DHB interfaces on the PCB. *OpenAgribot* is a new 'medium sized' platform in development for agricultural actuation or last mile delivery:



2 Design

The board, shown above, is designed to accommodate most typical use cases for wheeled, tracked and arm robots, including differential and Ackerman steering, open and closed loop control, and two low hardware levels of safety:

- **Dead Man's handle:** means that main power goes though a single high-quality switch connected by physical wire to a high-quality spring button held my a human operator. If this is released for any reason, all power is instantly cut.
- **Proximity sensors:** form the second level, such as the HC - SR04 ultrasonic sensor for measuring if any object gets in the near proximity. In that case it cuts out the power to the system as well. The board has direct connectivity provisions to the sensor mentioned.
- **Microcontroller:** Arduino Giga mounts a STM32H747XI dual Arm Cortex-M7+M4 32-bit low-power MCU. Giga has a Murata 1DX dual WiFi 802.11b/g/n transmitting data at 65 Mbps, and Bluetooth. It also has a camera interface.

It has 76 Digital I/O pins, of which 12 can be configured as Analog, and 12 as PWM. It has 2 on board DACs and 4 UART channels, 3 I2C channels, 2 SPI channels, CAN bus and audio jack. The CAN bus requires a transceiver (SN65HVD230) which is accomodated on the PCB. The PCB breaks out robust connectors for the CAN bus, I2C channels 1, 2 of the UARTs, and a 3.3V dual channel stereo DAC.

- **Robust connectors:** XT-60 (high-power) and JST-XH (low-power) are provided needed by practical vehicles in extreme environments, which vibrate and dislodge weak connections such as Arduino's duPont connectors. An Arduino Giga is soldered into the PCB, making its own connections robust but enabling easy reuse of the PCB between different robots.
- **Power Supplies:** The board is powered directly from 24V vehicle batteries. It provides 3 high power 24V outputs and 1 low power 24V output. The high power outputs can be used to step down the voltage and convert it to 12V and 5V using external buck converters which interface with the board's robust 12V and 5V connectors. These supplies are each fused on-board, on the input and output sides, using resettable, high reliability automotive grade ATC fuses. There are 3 low power and 1 high power 12V and 5V outputs. The board uses the 5V internally for its microcontroller. A precision referenced 3.3V supply is also provided which can supply a maximum current of 800mA in total. The PCB is designed for a temperature rise of 28C for a maximum current of 20A at 24V which is expected when used in OpenPodcar, the largest demonstrator platform.
- **Analog Input channels:** There are 6 Analog Input channels connected to the ADC of the Giga. These channels have a voltage divider circuit, which can be configured using the preset, to divide the incoming voltage. This allows a maximum input of 50V at each analog input channel. They are using the precision referenced 3.3V generated on board thus making the precision of readings very high and reliable, and are also zener clamped to protect from over voltage situations. 4 out of the 6 analog input channels have an optionally connectable jumper connection to measure the on board voltage signals on the power supply side.
- **Stepper motor channels:** The board has 4 stepper motor channels which can connect to stepper motor drivers such as Microstep M542. Stepper speed and direction is controlled by timed signals generated by a library running on the Giga along with a single enable signal for all 4 channels.
- **Quadrature encoder input channels:** The PCB also allows 4 Quadrature encoders to be directly interfaced. Quadrature inputs with 5 pins. which have an additional switch functionality are the ones used as a reference to design the interface.
- **BLDC motor driver interfaces:** There are 4 BLDC motor driver channels driven by a 4 channel 5V on board DAC (MCP4728). They allow individual speed, brake and direction control of the interfaced BLDC motors separately, with minimal noise.
- **Ultrasonic proximity sensor interfaces:** The board has separate 4 channels for interfacing ultrasonic sensors such as the HC - SR04 for detecting

objects in close proximity. Using these the second additional layer of security can be implemented as mentioned before.
- **MCP23017_SO GPIO extender:** It has two ports which perform different functionalities on board the PCB. Port A controls the direction and brake signals for the BLDC motor channels and Port B controls the 8 channels for the relay board interface mentioned below.
- **OSMC H-bridge channels:** The board provides interfacing channels for Open Source Motor Control (OSMC) drivers. This allows usability of a wide range of motors that can be driven using these controllers from 13V-50V and upto 160A max constant current ratings.
- **DHB H-bridge interfaces:** The reference used to design this interface is a DHB-12, which is a dual motor driver H-bridge that can control two 5V-15V motors at 30A constant current. The board has two such interfaces which allows four more motors to be controlled separately for low-medium power.
- **PCA9685PW I2C based LED controller:** There are 3 units present on board the PCB each performing a different task. The first one controls the OSMC driver channels. The second one controls the DHB H-bridge interfaces. The third one controls the servo channels mentioned below.
- **Servo channels:** The board has the capability of interfacing 16 standard RC servo motors. Due to this the PCB makes it possible to control anything that uses up to 16 servo motors such as robotic arms.
- **8 channel relay board interface:** This interface is used to control and switch an 8 channel relay board and control each relay signal separately due to the on board GPIO extender. These relay signals can be used to switch important signals like the Dead Man's handle relay signal or the ignition system on the OpenPodCar.
- **Mini screen display:** Connectivity and mounting are provided for an optional 3.5" TFT or LCD screen, useful for monitoring and diagnosing low-level faults, such as displaying low-level power and sensor information.

Fully-automated or manual build are both enabled by the design. Its CAD format and selected components are designed to be compatible with major pick-and-place builders such as `www.pcbway.com`. This enables new users to order an assembled PCB in just a few clicks. Alternatively, manual placing can be performed with a solder mask. The board is designed so that some features and components can be mounted optionally as per the use case to lower build cost.

Source files are available at `https://gitlab.com/charles.fox/r4pcb`

References

1. Alabi, E., Camara, F., Fox, C.: Evaluation of OSMC open source motor driver for reproducible robotics research. In: TAROS (2023)
2. Camara, F., Waltham, C., Churchill, G., Fox, C.: OpenPodcar: an open source vehicle for self-driving car research. Jounral of Open Hardware (in press; preprint arXiv:220504454)
3. Pearce, J.M.: Building research equipment with free, open-source hardware. Science **337**(6100), 1303–1304 (2012)

Naturalistic Robot Arm Trajectory Generation via Representation Learning

Jayjun Lee and Adam J. Spiers[0000−0002−3221−1000]

Imperial College London, London, UK
{jayjun.lee19,a.spiers}@imperial.ac.uk

Abstract. The integration of manipulator robots in household environments suggests a need for more predictable and human-like robot motion. This holds especially true for wheelchair-mounted assistive robots that can support the independence of people with paralysis. One method of generating naturalistic motion trajectories is via the imitation of human demonstrators. This paper explores a self-supervised imitation learning method using an autoregressive spatio-temporal graph neural network for an assistive drinking task. We address learning from diverse human motion trajectory data that were captured via wearable IMU sensors on a human arm as the action-free task demonstrations. Observed arm motion data from several participants is used to generate natural and functional drinking motion trajectories for a UR5e robot arm.

Keywords: Human-like Robot Motion · Self-Supervised Learning · Graph Representation Learning · Imitation Learning.

1 Introduction

For people with motion impairments, the ability to feed oneself is a major factor of independence [3]. Recently, wheelchair- or desk-mounted robotic manipulators have been implemented with these tasks in mind [1, 2]. In human-robot interaction (HRI), it has been observed that the human comfort and confidence may be increased by generating predictable and naturalistic motion paths [4, 9]. As such, we are aiming to add human-like arm motion to an assistive drinking task.

To generate human-like robot arm motion we collect human arm movement data using wearable IMUs. We then reconstruct action-free human arm trajectories to gain access to low-dimensional states, and use an autoregressive spatio-temporal graph neural network (GNN) to ingest this data in a self-supervised way. We learn internal model representations of human drinking dynamics that exploit the spatial and temporal relation between arm joints based on the Space-Time Separable Graph Convolutional Network (STS-GCN) [8]. By behaviour cloning (BC) from the human motion data collected via IMUs, we were able to generate diverse, human-like drinking robot arm motion that is functional across various bottle positions with heuristics to complete other subtasks in sequence.

In this work we have adapted the STS-GCN architecture from the human pose prediction community into an autoregressive GNN for self-supervised imitation learning for robotics, with the Mean Per Joint Position Error (MPJPE)

as the BC loss. As a result, the new system learns an internal model dynamics of naturalistic drinking motion with relatively sparse input data, making it suited to fewer (motion captured) demonstrations. The more compact result is also better suited for implementation on physical hardware with predictions further back in time making it suited to functional tasks.

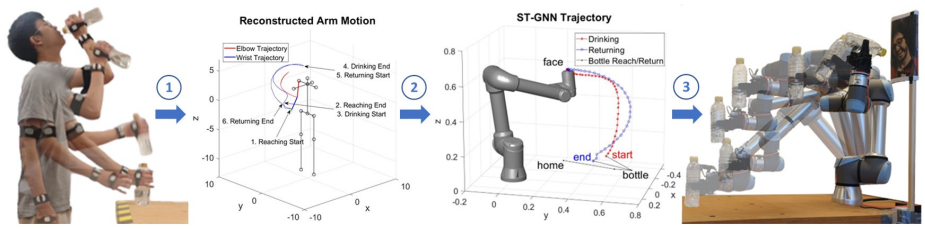


Fig. 1: Human drinking motion is captured using wearable IMUs, with the arm trajectory reconstructed to form an action-free demonstration. An autoregressive spatio-temporal GNN learns the motion dynamics from diverse drinking data to generate generalised naturalistic drinking motions which are scaled for a UR5e.

2 Related Work

2.1 Human-like Arm Motion Generation for Robots

It has been proposed that human-like behaviours of robotic manipulators can ensure safety, predictability, and social acceptance [4, 9]. Many research efforts have aimed for various aspects of human-like robot motion planning. One popular approach is movement primitives that decompose motion into a set of primitives that can be combined to generate complex movements and learned from human demonstrations [7]. We have adopted a self-supervised learning method that can generate diverse and generalisable human-like motion while learning an internal model dynamics with an autoregressive structure and without primitives. It is noted that our approach and MPs could potentially complement each other.

Human motion forecasting deals with the problem of predicting the 3D coordinates of V body joints for the future K frames, given past T frames. A skeleton-based model of human body may be used to form a graph structure, where each joint is a node [5, 6]. In [8], the STS-GCN model is introduced, which learns to encode the human body dynamics by factorising the spatio-temporal graph adjacency matrix to separate spatial and temporal adjacency matrices and focus on the joint-joint and temporal relations. We modify this architecture to learn from relatively sparsely logged data by extending the model to train autoregressively with a self-supervised loss. The result is a more compact learned internal model of human motion dynamics that can predict much further in time. We also take an embodied approach that maps the generated human arm trajectory onto a real robot arm to complete functional tasks, as opposed to visualisations of simulated skeleton models. Unlike in [8], the input trajectory segment to our system is not the initial frames of a continuous action, but rather a preparatory motion to reach and grasp a bottle prior to the generated movement of bringing the bottle to a user's mouth.

3 Methods

To collect human drinking motion data, 3 MetaMotionS+ IMUs *(MBientLab)* are attached using Velcro straps along the participant's right arm on the upper arm, forearm, and back of hand. Euler angles are logged at 100Hz and preprocessed to address discontinuities and noise. Five participants (two female, mean age of 23.2 years) each provided 10 drinking demonstrations for 6 discrete bottle positions on a 2-by-3 grid on a desk. Each recorded trajectory is discretely down-sampled to 150 samples and split into reaching, drinking and returning phases in Fig. 1.

The GNN is shown in Fig. 2. The encoder has four STS-GCN layers with the input graph of $T = 30$, which learns the adjacency matrix of the input to highlight certain space-time edges with feature graphs. The decoder has five TCN layers to generate the output graph of $K = 30$ with 3D joint coordinates. The learned graph representations act as the internal model for the drinking dynamics to generate the subsequent motion segment given the input segment. The model is trained to minimise the MPJPE loss in Eq. 1 between the autoregressively generated 120-frame drinking trajectory and the ground-truth self-supervised label. This requires a recursive forwarding of its output to its input four times.

$$L = \frac{1}{VK} \sum_{k=T+1}^{T+K} \sum_{v=1}^{V} \|\hat{x}_{vk} - x_{vk}\|_2 \tag{1}$$

$x_{vk}, \hat{x}_{vk} \in R^3$ are the true and predicted joint v positions at frame k. V is the number of nodes per frame. T and K are the number of input and output frames.

As the generated human trajectory resides in the human workspace, we map and linearly scale the human wrist 3D trajectory to the robot arm's end-effector (EE) workspace, safely reaching the user's mouth. Future work would integrate sensing solutions to deal with the user moving and force-interactions.

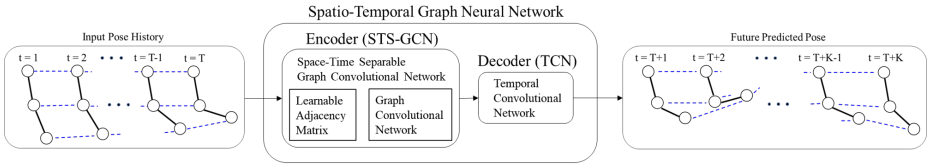
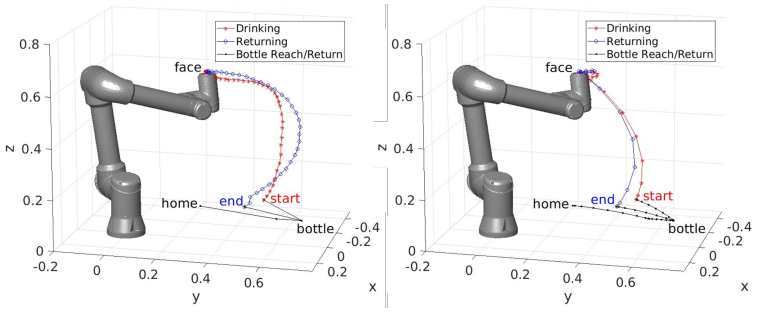


Fig. 2: An overview of the autoregressive GNN adapted from the STS-GCN.

4 Results

A 6 DOF UR5e robot arm was used with a parallel jaw gripper adapted from the ROBOTIS Open-Manipulator X robot, Fig. 1. In Fig. 3 we compare the GNN trajectory with a typical joint-space IK trajectory. Pronounced curves with hysteresis are present in the GNN trajectory. Such hysteresis also appears in human reaching motions [9]. We also test our trained model on unseen test bottle positions placed within the aforementioned 2-by-3 grid of bottle positions.



(a) GNN trajectory. (b) Joint-space trajectory.

Fig. 3: A comparison between a drinking trajectory generated by the GNN (a) and by joint-space inverse kinematic interpolation with trapezoidal velocity profile. (b) Note that a classic task space IK trajectory would be a straight line.

5 Conclusion and Future work

We have proposed a preliminary GNN-based self-supervised imitation learning framework, using human demos to generate human-like robot arm drinking motion from a reach-to-grab motion. In future work, this could be extended by multi-task learning and with a camera to observe scene obstacles for more Activities of Daily Living where human-like motion is beneficial for assistive robots.

References

1. Beaudoin, M., Lettre, J., Routhier, F., Archambault, P.S., Lemay, M., Gélinas, I.: Impacts of Robotic Arm Use on Individuals with Upper Extremity Disabilities: A Scoping Review. Canadian Journal of Occupational Therapy (2018)
2. Chi, M., Yao, Y., Liu, Y., Zhong, M.: Recent Advances on Human-Robot Interface of Wheelchair-Mounted Robotic Arm. Recent Patents on Mechanical Engineering (2019)
3. Chung, C.S., Wang, H., Cooper, R.A.: Functional assessment and performance evaluation for assistive robotic manipulators: Literature review. Journal of Spinal Cord Medicine (2013)
4. Gulletta, G., Erlhagen, W., Bicho, E.: Human-Like Arm Motion Generation: A Review. Robotics (2020)
5. Jain, A., Zamir, A.R., Savarese, S., Saxena, A.: Structural-RNN: Deep learning on spatio-temporal graphs. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
6. Mao, W., Liu, M., Salzmann, M.: History Repeats Itself: Human Motion Prediction via Motion Attention. In: European Conference on Computer Vision (ECCV) (2020)
7. Paraschos, A., Daniel, C., Peters, J., Neumann, G.: Using probabilistic movement primitives in robotics. Autonomous Robots **42**, 529–551 (2018)
8. Sofianos, T., Sampieri, A., Franco, L., Galasso, F.: Space-Time-Separable Graph Convolutional Network for Pose Forecasting. In: IEEE International Conference on Computer Vision (ICCV) (2021)
9. Spiers, A., Khan, S.G., Herrmann, G.: Biologically inspired control of humanoid robot arms: Robust and adaptive approaches. Springer International Publishing (2016)

Receding Horizon Contact Planning for Advanced Motions in Hexapod Robots*

Daniel Stephen Johnson[0000−0001−9002−4257], Bruno Vilhena Adorno[0000−0002−5080−8724], and Simon Watson[0000−0001−9783−0147]

Manchester Centre for Robotics and AI, University of Manchester, Oxford Rd, Manchester M13 9PL, UK
daniel.johnson-2@manchester.ac.uk,
bruno.adorno@manchester.ac.uk, simon.watson@manchester.ac.uk

Abstract. This work proposes a planning approach for advanced motions in hexapod robots. Contact sequences are explored until a user-defined planning horizon is reached. The contact change leading to the best position on the horizon is executed, and exploration resumes. In preliminary simulations, the algorithm consistently returned paths requiring at least 20% fewer contact changes than the state-of-the-art. Our algorithm generated 48.0% fewer nodes than the state-of-the-art in one terrain and 0.5% more in another, leading future work to examine the effects of specific environmental features on the number of nodes generated.

Keywords: Contact Planning · Legged Motion Planning · Hexapod Robots.

1 Introduction

Legged robots present unique challenges for motion planning, given their high degrees of freedom, under-actuation, and the need to maintain balance [5]. While gaited walking methods suffice in many cases [4], environments such as that shown in Fig. 1 require robots to plan and execute advanced forms of motion in order to navigate.

To plan such motions, legged robots must identify where they will make contact with the environment and the sequence in which contacts are made or broken. This work proposes a novel contact planning algorithm incorporating receding horizon methods to plan advanced hexapod motions. The planner is tested against state-of-the-art using a simulation of the Corin hexapod [1].

2 Planning Algorithm

Our Receding Horizon Contact Planning (RHCP) algorithm is based on the Contacts Very Best First Planning (CVBFP) approach of Escande *et al.* and shares several features in common with it, including the same potential field, guide path, and posture generator (PG) [3]. A flowchart summarising RHCP is shown in Fig. 2. The planner uses a tree search to explore possible stance sequences, where a "stance" refers to a set

* This work was supported by a grant from the University of Manchester.

Fig. 1: An example of an environment requiring advanced motions to navigate. In this case, chimney walking (*left*) and wall walking (*right*) [2]

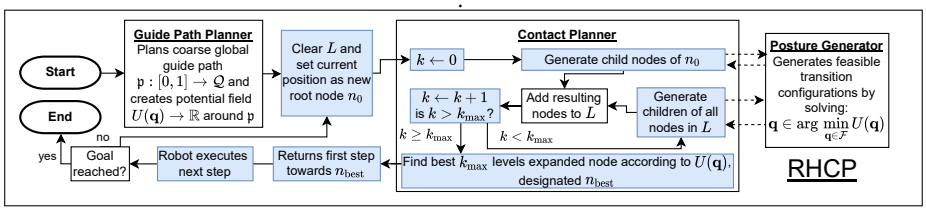


Fig. 2: Flowchart showing the operation of the RHCP algorithm. Elements present in RHCP but not in CVBFP are shown in *blue*.

of contacts made simultaneously, an example of which is shown in Fig. 3. Each node in the search is associated with a stance and a transition configuration that allows the robot to move into that stance from the previous stance. When these transition configurations exist, they are found by the PG, shown in Fig. 2.

RHCP explores possible stance sequences until a user-defined maximum sequence length $k_{\max}$ is reached. As these sequences always begin from the robot's current stance, the planning horizon effectively recedes each time the robot moves and its stance changes. The planning process starts with the root node n_0 being expanded, generating child nodes as follows:

- one child node is generated for each foot in contact in n_0 that could be lifted;
- for each foot not in contact in n_0, one child node is generated with that foot placed on each surface within reach at a point chosen by the PG.

Each child node of n_0 is then expanded, producing a 2nd generation of children. This process repeats until $k_{\max}$ generations have been produced. The $k_{\max}$ generation node with the lowest potential $U(\mathbf{q})$ is found, and the first contact change in the sequence leading to that node is executed. The planning process repeats with the robot's new position replacing n_0 as the root node. This continues until the goal is reached.

3 Preliminary Results

RHCP (with $k_{\max} = 2$) and CVBFP were each used five times to plan a path for the Corin hexapod across a section of rough terrain, as well as a more basic environment

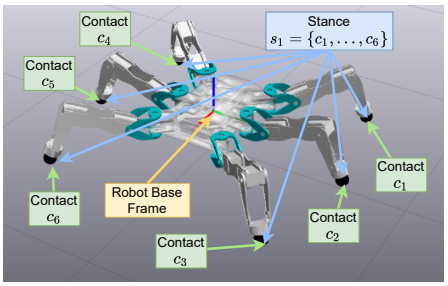


Fig. 3: Example of a stance with its constituent contacts labelled.

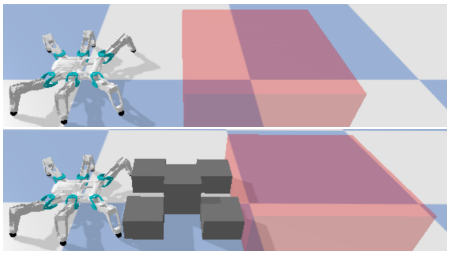


Fig. 4: Basic (*top*) and Rough (*bottom*) environments used for preliminary testing. Corin is shown in its starting configuration, and the goal region is shown in red.

without obstacles (shown in Fig 4). The testing results are summarised in Table 1. Two tests were cancelled (one of CVBFP in the basic environment one of RHCP in the rough environment) after they generated over 6000 PG calls without completion. These tests are excluded from Table 1 results.

Table 1: Planning data for CVBFP and RHCP in environments tested.

Environment	Algorithm	Calls to PG	Nodes	Distance Covered (m)	Contact Changes
Basic (no obstacles)	CVBFP	931	700	1.46	34
	RHCP	484	342	1.50	27
Rough (with obstacles)	CVBFP	2726	1392	2.25	55
	RHCP	2741	1350	2.16	30

As shown in Table 1, the average number of PG calls in the rough environment was just 0.5% higher for RHCP than for CVBFP. In the basic environment, however, RHCP made an average of 48.0% fewer calls to the PG than CVBFP.

Fig. 5 shows a plot of the potential of the nodes generated during the CVBFP tests in the basic environment. It can be seen on the graph that the average potential plateaus for several independent tests at a value of approximately 250. As this plateau is not observed in the RHCP tests on the basic environment, we believe that this is the principal reason RHCP made fewer PG calls than CVBFP. This is also believed to be why the two tests excluded from the results in Table 1 failed to conclude. We hypothesise that this plateau occurs because the robot has reached a state in which a foot that is critical for balance must be lifted to progress. An example of such a configuration is shown in Fig. 6. Future work will aim to confirm this hypothesis and understand what environmental features cause the algorithms to encounter this problem.

Table 1 also shows that the paths generated by RHCP required fewer contact changes than those generated by CVBFP, requiring 21.2% and 45.2% fewer in the basic and rough environments, respectively.

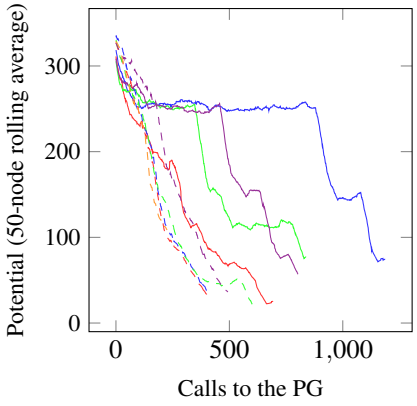


Fig. 5: Mean potential of nodes generated by the RHCP tests (*dashed lines*) and CVBFP tests (*solid lines*) in the basic environment.

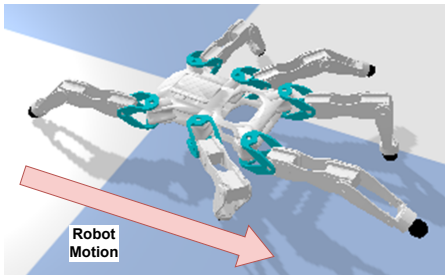


Fig. 6: Example configuration generated during planning plateau encountered by CVBFP in the basic environment. In the configuration shown, the robot must lift its right hind leg to progress, but doing so would cause it to tip backwards.

4 Conclusions

This work presented a novel receding horizon contact planner. In preliminary tests, the paths generated by RHCP required at least 20% fewer contact changes on average than those by CVBFP. Additionally, RHCP generated 48.0% fewer nodes than CVBFP in the basic environment while generating only 0.5% more nodes than CVBFP in the rough environment. Understanding which environmental features cause performance problems in the two algorithms is the subject of ongoing work.

References

1. Corin: Mobile hexapod for remote inspection and object manipulation (04 2022), https://uomrobotics.com/robots/corin.html
2. Cheah, W., Khalili, H.H., Arvin, F., Green, P., Watson, S., Lennox, B.: Advanced motions for hexapods. International Journal of Advanced Robotic Systems **16** (3 2019). https://doi.org/10.1177/1729881419841537
3. Escande, A., Kheddar, A., Miossec, S., Sylvain, M.: Planning contact points for humanoid robots. Robotics and Autonomous Systems **61**, 428–442 (2013). https://doi.org/10.1016/j.robot.2013.01.008ï, https://hal-lirmm.ccsd.cnrs.fr/lirmm-00800829
4. Silva, M.F., MacHado, J.A.: A literature review on the optimization of legged robots. JVC/Journal of Vibration and Control **18**, 1753–1767 (10 2012). https://doi.org/10.1177/1077546311403180
5. Wieber, P.B., Tedrake, R., Kuindersma, S.: Modeling and Control of Legged Robots, pp. 1203–1234. Springer International Publishing, Cham (2016), https://doi.org/10.1007/978-3-319-32552-1_48

Object Feature Reconstruction via Robotic In-Hand-Manipulation and Haptic Proprioception

Ciaran Dowds, Rafaela Sampaio, Adam J. Spiers

Manipulation and Touch Lab, Department of Electrical and Electronic Engineering, Imperial College London, Exhibition Rd, South Kensington, London, UK

Abstract. In this extended abstract we demonstrate that within-grasp object localisation and size estimation can be achieved without tactile sensors, using only a simple 3DOF robotic manipulator and proprioceptive sensing from inexpensive actuators. A geometry-based algorithm is used, based on 4-bar linkage kinematics, modified to match the structure of our gripper when holding a convex object, such as a cube.

1 INTRODUCTION

In robotics, the ability to determine object and grasp properties by touch has been investigated for many years. More recently, it has been demonstrated that in-hand-manipulation (IHM) can be used to enhance tactile data collection, by exposing more of an object's surface to tactile sensor surfaces [2, 5]. The potential for tactile object identification without tactile sensors was explored in [4], where time series actuator position and current was reduced to several key points and fed to a classification algorithm. In this work we go beyond object classification and aim to estimate physical parameters of an object's size and pose within the gripper. Our method estimates object location and size based on a kinematic representation of the hand-object model as a four-bar linkage whose parameters can be estimated using the Freudenstein equation [1].

2 IMPLEMENTATION

2.1 Gripper

To focus on the algorithmic challenges of the in-hand proprioception problem, we focus our efforts on planar manipulation. We make use of a new gripper in this work, called the MinE-TRoll, which is a miniature and simplified version of the open-source E-TRoll (Extended-Tactile Rolling) gripper of [5].

The gripper consists of two rigid fingers with a single revolute joint at the base of each. The two revolute joints are mounted on a prismatic dual-rack-and-pinion joint, which enables their linear separation to be adjusted.

Two types of manipulation are used to gather proprioceptive data about the held object. These are a fixed-base rolling manipulation (previously used in [4, 2, 3]) and a new manipulation strategy which we call 'palm-pivot'. These manipulations are executed by the gripper sequentially, following an initial grasp (Fig 1).

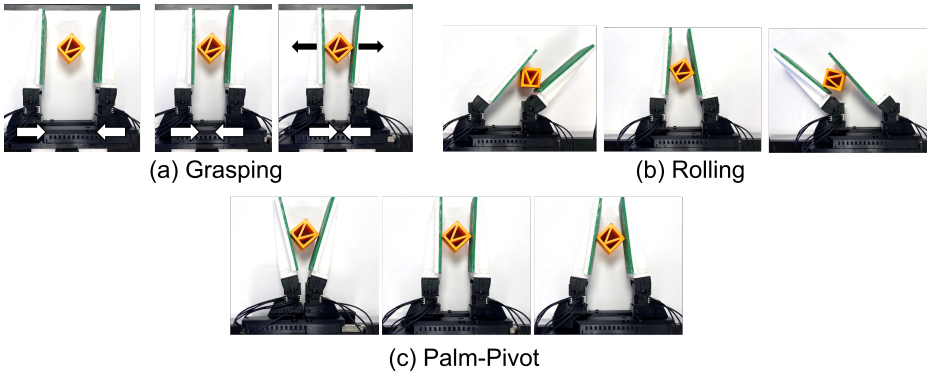


Fig. 1. The 3 in-hand-manipulations undertaken by the MinE-Troll gripper.

2.2 Kinematic Model and Parameter Solver

The relationship between the *MinE-TRoll* fingers and an object can be kinematically modelled as a four-bar linkage mechanical system with parameters as shown in Fig 2. The palm width aperture is represented by a, and c is the estimated object size. Angles ψ and ϕ represent the angles, and b and d the distances, from the centre of the finger joints to the object contact points. The relationship 1 between the parameters is derived from Freudenstein's Equations [1].

$$R_1 \cos(\phi) - R_2 \cos(\psi) + R_3 - \cos(\phi - \psi) = 0 \tag{1}$$

where,

$$A = (1+R_1)\cos(\phi)+R_2+R_3, B = -2\sin(\phi), C = (R_1-1)\cos(\phi)-R_2+R_3 \tag{2}$$

$$R_1 = \frac{a}{d}, R_2 = \frac{a}{b}, R_3 = \frac{a^2+b^2-c^2+d^2}{2bd} \tag{3}$$

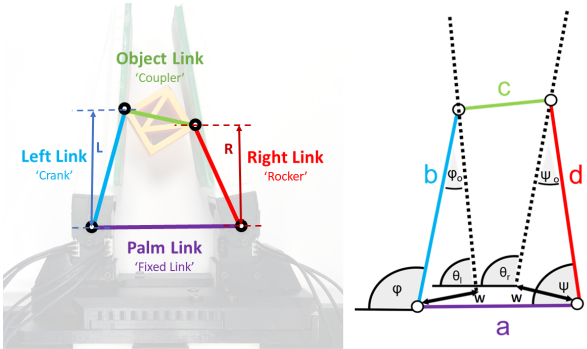


Fig. 2. The relationship between the fingers and object may be represented by a four-bar mechanism, assuming point contacts.

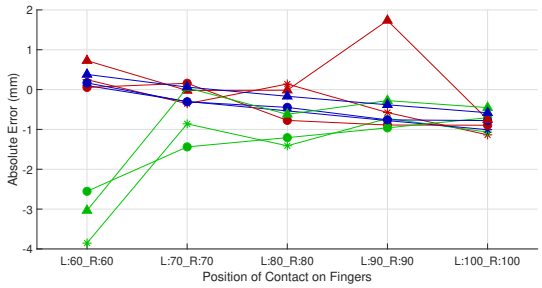


Fig. 3. Object size estimation error for different manipulations and object sizes with symmetrical initial contact points. Note that a legend would not fit on this plot but is the same as in Fig 4.

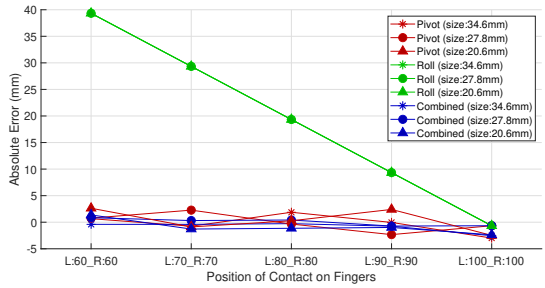


Fig. 4. Left finger contact point estimation error with symmetrical initial contact points.

Due to the offset w caused by the width of the finger, there is no direct measurement of angles ϕ and ψ. The measured joint angles θ_l and θ_r are related to ϕ and ψ by equation 4.

$$\phi = \theta_l + \arcsin(w/b) \quad \psi = \theta_r - \arcsin(w/d) \tag{4}$$

Observations of a, θ_l and θ_r (from the actuator encoders) during IHM are substituted into the above equations to form a vector of nonlinear functions with each component in the form $f_i(a_i, \theta_{l_i}, \theta_{r_i}, b, c, d) = 0$. MATLABs nonlinear least squares solver was used to find a set of parameters b,c,d which minimise the sum of squares of the component residuals. These parameters are optimal estimations of the contact point locations and proximity.

3 RESULTS

The manipulation test procedure involved two sequential phases of motion, a $\pm 15mm$ palm-pivot manipulation followed by a $\pm 10°$ rolling manipulation. Parameter estimation was performed using data from each phase of the manipulation separately and combined.

Figure 3 shows the absolute error for object size estimation for different (symmetrical) initial contact positions on the fingers and object sizes. Figure 4

shows the absolute error for contact points estimation on the left finger (the plot for the right finger is not shown but is very similar). The results shown also consider the different manipulation strategy used to generate the data, red colors represent palm-pivot actions, green series represent rolling actions, and blue colors represent the combined estimation considering both actions. Better size estimation is achieved for the combined pivoting and rolling data rather than either individually. Size estimation error with combined data is less than $\pm 1mm$. Position estimation is unacceptable with rolling data alone, with combined data giving marginal improvements over pivoting alone. Position estimation error with the combined data is approximately $\pm 5mm$.

4 CONCLUSION AND FUTURE WORK

This work has introduced a kinematic model based parameter estimation algorithm that uses only actuator encoder data to reconstruct the physical properties of an object being manipulated. We demonstrate the approach on practical data collected from the MinE-TRoll, a custom-built and inexpensive 3DOF robotic gripper. The 3D-printed nature of the gripper, inexpensive actuators (Dynamixel XC330-T288-T) and slipping of objects against the fingers introduces various forms of noise and positioning error. Despite this our algorithm performs well, giving object size estimation error within 5% of an object's size.

Key to exploiting the algorithm is the use of multiple In-Hand-Manipulation actions during data collections. It was demonstrated that combining typical rolling manipulation with a new palm-pivot action greatly reduces size and position prediction error. This pivoting action is made possible by the variable width palm of the MinE-TRoll robotic gripper. In future work we will investigate the effect of varying palm width also for rolling motions, to keep the fingers parallel during this action, as in [5]. We also plan to expand the objects under examination to include those with curved and irregular cross-sections. Finally, the relationship between object starting position and overall accuracy may also be further studied, as this parameter is bound to vary in practical applications.

References

1. Freudenstein, F.: Design of four-link mechanisms (1954)
2. Mohtasham, D., Narayanan, G., Calli, B., Spiers, A.J.: Haptic object parameter estimation during within-hand-manipulation with a simple robot gripper. In: 2020 IEEE Haptics Symposium (HAPTICS) (2020)
3. Spiers, A.J., Calli, B., Dollar, A.M.: Variable-friction finger surfaces to enable within-hand manipulation via gripping and sliding. IEEE Robotics and Automation Letters (2018)
4. Spiers, A.J., Morgan, A.S., Srinivasan, K., Calli, B., Dollar, A.M.: Using a Variable-Friction Robot Hand to Determine Proprioceptive Features for Object Classification during Within-Hand-Manipulation. IEEE Transactions on Haptics (2020)
5. Zhou, X., Spiers, A.J.: E-troll: Tactile sensing and classification via a simple robotic gripper for extended rolling manipulations. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2022)

Detection and Classification of Unmanned Aerial Vehicles using Commercial Off-The-Shelf Radar Detectors

Christopher Wheat, Jonathan M. Aitken

Department of Automatic Control & Systems Engineering, University of Sheffield
{cdwheat1, jonathan.aitken}@sheffield.ac.uk

Abstract. Obtaining key information about a malicious UAV such as position and the vehicle type is vital to protect the airspace around critical infrastructure. Radar offers a means of obtaining this information, with each UAV presenting a unique micro-Doppler signature caused by propeller rotation and airframe vibration. This paper outlines a system for detecting and clasifying UAVs and giving an indication of payload.

Keywords: UAV · Radar · UAV Detection · Micro-Doppler · CNN

1 Introduction

With the rise of ever smaller and affordable system-on-chip (SOC) components, Unmaneed Aerial Vehicles (UAVs) have become increasingly accessible, with predicted sales in the US going from 600,000 in 2016 to 2.7 million in 2020[1]. Combating this growing threat has led to the development and research of UAV detection and elimination, with commercial systems often deployed at critical infrastructure e.g. the Elvira 360° System [1]. Despite this, much critical infrastructure still goes unprotected [2], especially in remote sits, largely due to the impractical cost of deploying these commercial systems at the necessary scale.

2 System Concept and design

Figure 1 shows a concept of operations diagram with multiple radar detector systems in place around critical infrastructure, each capable of scanning the airspace within their vicinity. Research has proven the effectiveness of classification using the radar returns received from UAVs using deep learning methods [3–5]. As a result of automated classification, information regarding the detected UAV type and size can also be provided to the user.

The final system is shown in Figure 2 that shows all hardware used, with this all interacting as seen in the functional architecture in Figure 3.

[1] https://www.faa.gov/newsroom/faa-releases-2016-2036-aerospace-forecast?newsId=85227

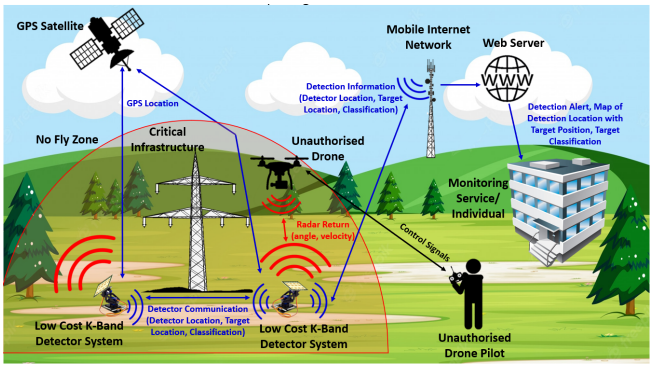


Fig. 1: Concept of Operation of the UAV Detector

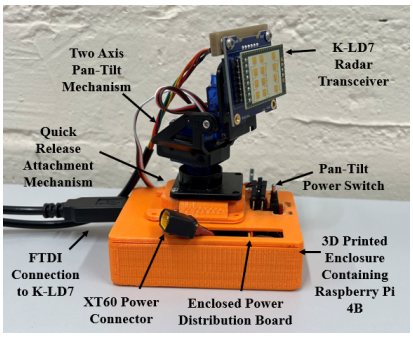


Fig. 2: Final System

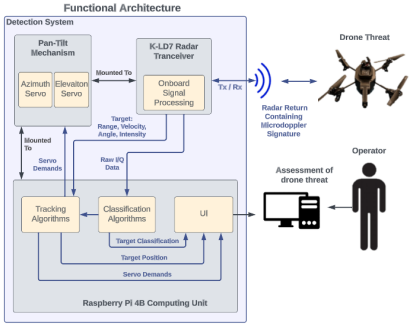


Fig. 3: System Architecture

2.1 UAV Classification

A breakdown of the Target Classification process involved to classify the UAV is shown in Figure 4. This uses a spectrogram derived from the radar sensor, an example of a spectrogram for a DJI Phantom is shown in Figure 5. The vibrations in the spectrogram are produced by both the baldes and the body. Drone design and selection of motor speed, propellor size and pitch is closesly related to the payload capacity [6].

3 Methodology, Results and Discussion

Four different UAVs (5-inch racing drone, Parrot Ar.Drone, DJI Phantom 4 and DJI Inspire 1) are used to train the deep learning model in addition to a "no drone" class. This data was generated in a flying lab at the University of Sheffield. Training took place at a range of different ranges, and validation models were compared using representative data sets across different ranges of the sensor.

A model has then been trained on the two extremes of the range data gathered, 0 - 4, and 7 - 11 m. When compared to the model trained against all ranges,

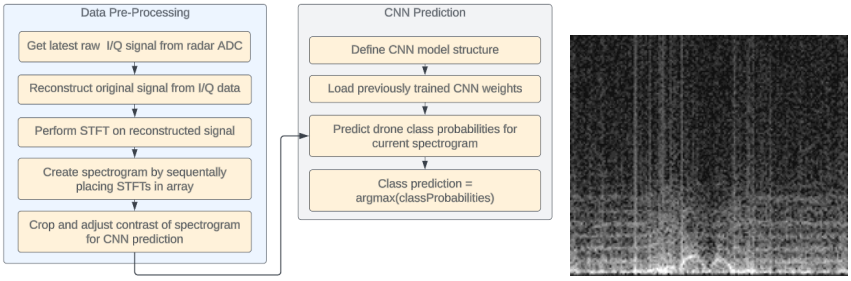


Fig. 4: Classification Process using a spectrogram created form Short-Time Fourier Transform (STFT) of radar returns

Fig. 5: DJI Phantom Spectrogram

Ranges Trained	Ranges Validated	Training	Validation
0 - 4 m	0 - 4 m	100%	96.43%
0 - 4 m	0 - 4 m, 3 - 7 m, 7 - 11 m	100%	65.49%
0 - 4 m, 3 - 7 m, 7 - 11 m	0 - 4 m, 3 - 7 m, 7 - 11 m	97.22%	89.28%
0 - 4 m, 7 - 11 m	0 - 4 m, 3 - 7 m, 7 - 11 m	98.65%	87.52%

Table 1: Effects of Range used for Training and Validation on Model Accuracy

the model validated only at the extremes of UAV range suffers less than a 2% drop in accuracy (Figure 6. Therefore, the majority of features used to classify the spectrograms can be extracted without training on the data obtained at the middle 3 - 7 m range, as shown in Table 1.

4 Conclusions

A prototype system has been constructed, with custom software and algorithms implemented. Tracking of UAVs is performed by a proportional controller using target data obtained from the onboard signal processing of the K-LD7. This allows the positional information of the UAV relative to the radar to be acquired.

Meanwhile classification is performed by a CNN based on previously successful models from research. This CNN has been trained on a custom dataset generated on from real UAVs at varying ranges, with the impact investigated. Producing a model with good classification accuracy over multiple ranges is necessary in a real system due to the unpredictable nature and flight path of unauthorized UAV flights. The final model is capable of classifying UAVs at ranges of 0 - 11 m with an accuracy of 89.28% at 1 Hz. The sensor selected for this project operated over a short range (maximum distance 15m for a person). This paper details the process for using the sensor for detection and classification of UAVs, this can be extended to compatible longer range sensors.

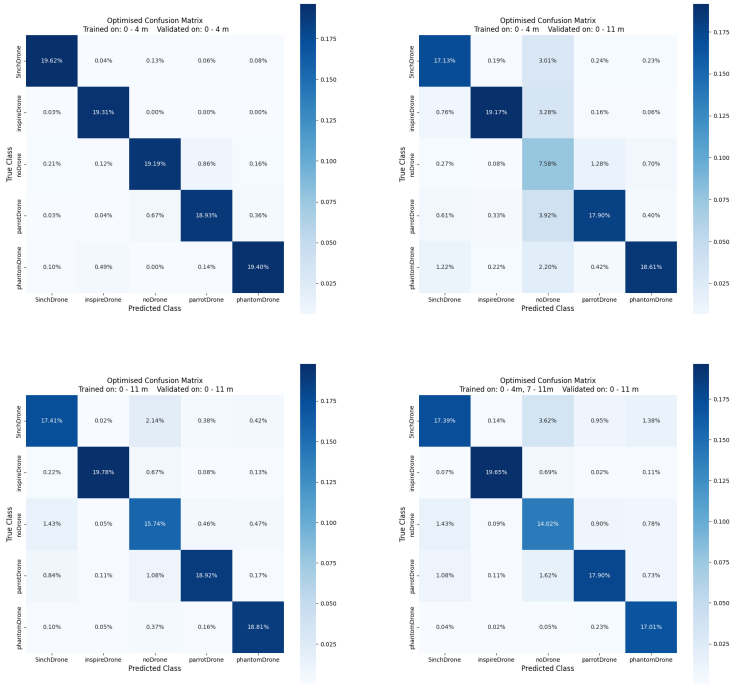


Fig. 6: Confusion matrices showing individual class prediction accuracy for models trained and validated against data gathered at different ranges.

References

1. Ioannis K Kapoulas, Antonios Hatziefremidis, AK Baldoukas, Evangelos S Valamontes, and JC Statharas. Small fixed-wing uav radar cross-section signature investigation and detection and classification of distance estimation using realistic parameters of a commercial anti-drone system. *Drones*, 7(1):39, 2023.
2. Xindi Zhang and Krishna Chandramouli. Critical infrastructure security against drone attacks using visual analytics. In *International Conference on Computer Vision Systems*, pages 713–722. Springer, 2019.
3. Samiur Rahman and Duncan A Robertson. Radar micro-doppler signatures of drones and birds at k-band and w-band. *Scientific reports*, 8(1):17396, 2018.
4. Albert Huizing, Matijs Heiligers, Bastiaan Dekker, Jacco de Wit, Lorenzo Cifola, and Ronny Harmanny. Deep learning for classification of mini-uavs using micro-doppler spectrograms in cognitive radar. *IEEE Aerospace and Electronic Systems Magazine*, 34(11):46–56, 2019.
5. Bilal Taha and Abdulhadi Shoufan. Machine learning-based drone detection and classification: State-of-the-art in research. *IEEE access*, 7:138669–138682, 2019.
6. Spyridon G Kontogiannis and John A Ekaterinaris. Design, performance evaluation and optimization of a uav. *Aerospace science and technology*, 29(1):339–350, 2013.

Towards Lifelong Social Robot Navigation in Dynamic Environments*

Qi Zhang[1], Ioannis Stefanakos[1], Javier Cámara[2], and Radu Calinescu[1]

[1] Department of Computer Science, University of York, York, UK
[2] ITIS Software, Universidad de Málaga, Málaga, Spain

Abstract. We describe a work-in-progress approach to solving the problem of robot navigation in dynamically changing, social environments. Our approach employs reinforcement learning informed by a continually updated model that predicts the evolution of the environment, and handles two common scenarios: (1) a person moving within the environment, and (2) static obstacles with positions that change over time. We assess the effectiveness of the approach in a simulated assistive-care application in which a mobile robot supports a person with mild cognitive or physical impairments with simple everyday tasks.

Keywords: Reinforcement Learning · Navigation · Self-Adaptive Systems · Assistive-Care Robots.

1 Introduction

Autonomous robots have been increasingly deployed alongside humans, operating in complex environments and in various domains such as assistive-care [5], manufacturing [14], and nuclear fusion [3], among others. Assistive-care robots [4] are emerging as a vital tool for providing care and support to the elderly in their homes [6]. These robots are designed to perform tasks that include retrieving objects, aiding in mobility, monitoring health metrics, and offering medication reminders [15].

Machine learning (ML) algorithms have enabled a personalized assistance offered by these robots through their adaptation to the different challenges and preferences of users [10], fostering a higher level of independence for older adults. However, an important challenge still largely unaddressed is ensuring lifelong performance of ML (i.e., the ability to adapt to changes during long-term assistance) in assistive robotics for elderly care [9]. Navigating in complex and constantly changing environments presents potential hazards to human-robot interactions. For instance, an unexpected collision between the robot and a human could lead to the person falling. Thus, ensuring that assistive-care robots maintain a high-level of effectiveness and adaptability during long-term assistance is of utmost importance. This includes the ability to avoid collisions with humans moving within the environment and obstacles whose positions may change from

* Supported by Assuring Autonomy International Programme.

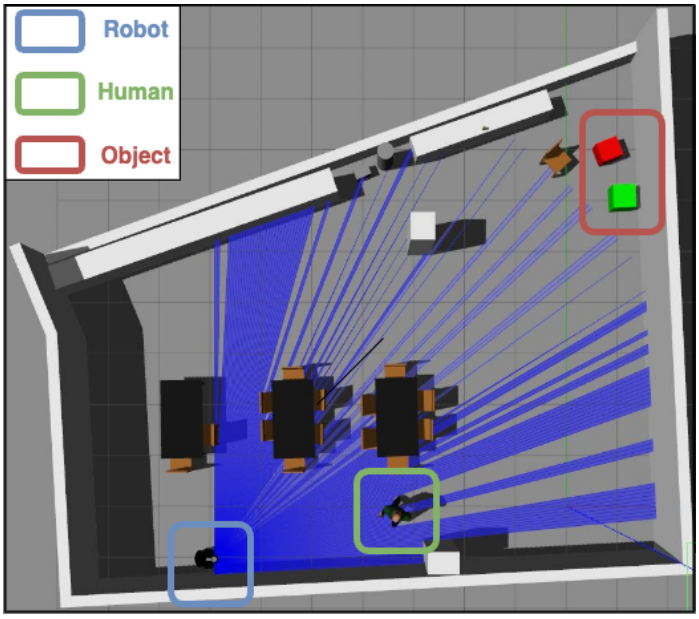


Fig. 1. View of a large open plan kitchen area in Gazebo[3] simulation environment.

time to time (e.g., furniture). Moreover, by achieving lifelong performance, the robots can provide consistent service to people with mild cognitive and/or motor impairments whose condition may evolve over time, resulting, for instance, in changes in movement patterns (erratic trajectories, speed variation, etc.).

In this work-in-progress study, we aim to enhance the navigation capabilities of a robot within a dynamic environment where human paths and object locations change over time. Solving this problem presents challenges, such as anticipating human trajectories, and proactively adapting to changing environmental conditions before undesirable situations are given (e.g., those that entail a high risk of collision with people and objects). To overcome these challenges, we propose an approach that employs reinforcement learning to endow a robot with the ability to reach a target location within the environment while avoiding collisions both with static and dynamic obstacles.

2 Navigation in Dynamic Environments

Figure 1 depicts the environment that we employed to evaluate our approach, which corresponds to a large open plan kitchen area with various static obstacles, such as chairs and tables. In a healthcare assistance scenario, the trajectories of human movement can often be unpredictable (e.g., erratic trajectories

[3] https://gazebosim.org/home

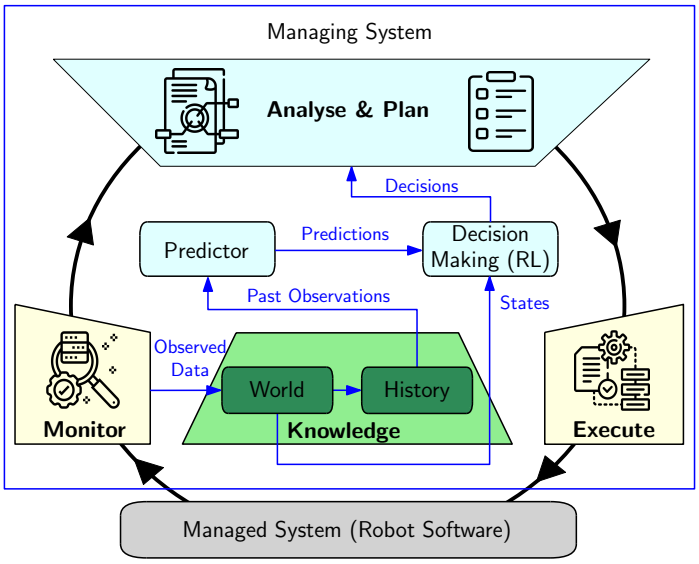


Fig. 2. MAPE-K Loop via Reinforcement Learning.

derived from cognitive and motor impairments) and static obstacles may frequently change location due to situations such as a patient forgetting to return items to their original locations. The co-existence of these two factors can pose challenges for existing navigation algorithms to handle effectively. A human participant serves as a moving obstacle, with changing motion patterns (walking speed, direction) to which the robot has to adapt at runtime in order to reach the goal position while avoiding collisions. To simulate cluttered environments commonly found in elderly care settings, the position of static obstacles, such as chairs, changes over time, mimicking real-world scenarios where a person may leave clothing on the ground or rearrange the furniture from time to time.

3 Overview of the Approach

Our approach employs MAPE-K [7], which is regarded as one of the most successful paradigms to build autonomic and self-adaptive systems. MAPE-K consists of four stages arranged in a feedback loop (Monitor, Analyse, Plan, Execute) and a Knowledge base. As illustrated in Figure 2, the MAPE-K loop in our approach is supported by a reinforcement learning framework, which is incorporated in the software running in the robot. The Monitor stage focuses on gathering world data, which includes the robot's distance to obstacles and the coordinates of the human in the environment, and incorporates it into the World model. Simultaneously, the world observations are incorporated into the History model, which is tasked with preserving past observations to be consumed by the Predictor component. The Predictor component is integrated into the loop to predict

the human state within a short time horizon, including its trajectory. World data about current state obtained from the World model, along with predictions from the Predictor component, serve as inputs to the Decision Making reinforcement learning (RL) component. These inputs include the distance to obstacles, target location information, values from the action space, and predicted human coordinates. Subsequently, the set of actions generated as output by the RL component are set for execution. In our instantiation of MAPE-K, the Analyse and Plan stages are unified into a single stage supported by the tandem operation of the Decision Making and Predictor components.

4 Related Work

Significant research has been conducted on robot navigation and path prediction in various environments. The study in [8] examines a robot's ability to navigate among humans in a manner that adheres to social norms and ensures safety. However, it does not account for changes in the position of static obstacles over time, and the performance diminishes with increasing map size. A path planning approach, introduced in [11], identifies paths in either static or dynamic cluttered environments, but is primarily applied within a grid-based model framework without real-world scenarios. The use of occupancy values from presampled trajectories as part of the observation space is highlighted in [1] as an effective way to reduce training time for human path prediction. The authors plan to adjust their reward function for more complex obstacle scenarios. A hybrid online planning approach for navigation in hospital-like settings is presented in [13], with plans to further develop this approach by incorporating a learning-based strategy. While previous research on navigation in dynamic environments is often focused on adapting to dynamic obstacles such as humans, it frequently overlooks changes in static obstacle locations and the prediction of human paths, which are crucial for maintaining safety in scenarios with potential erratic human behaviour due to mild cognitive and motor impairments.

5 Conclusion

In this paper, we present a method for lifelong navigation based on a dynamic and cluttered environment. Within the framework of a self-adaptive system, the navigation task is formulated as a reinforcement learning process. In such a way, the robot acquires the ability to navigate towards a designated goal while avoiding both moving humans and obstacles that change position over long time.

In future work, we aim at enhancing the method's ability to predict human movement paths at runtime, provide safety guarantees (e.g., minimising probability of collisions against obstacles) by complementing RL with quantitative verification techniques [2,12], and explicitly considering trade-offs among multiple qualities (e.g., level of disruption to the human vs. usefulness in task assistance). We also plan to broaden the range of scenarios to assess the generality of our approach and evaluate how it performs under a diverse set of situations.

References

1. Akmandor, N.U., Li, H., Lvov, G., Dusel, E., Padir, T.: Deep reinforcement learning based robot navigation in dynamic environments using occupancy values of motion primitives. In: Intelligent Robots and Systems. pp. 11687–11694 (2022). https://doi.org/10.1109/IROS47612.2022.9982133
2. Casimiro, M., Garlan, D., Cámara, J., Rodrigues, L., Romano, P.: A probabilistic model checking approach to self-adapting machine learning systems. In: Software Engineering and Formal Methods. vol. 13230, pp. 317–332 (2021). https://doi.org/10.1007/978-3-031-12429-7_23
3. Devlin-Hill, B., Calinescu, R., Cámara, J., Caliskanelli, I.: Towards scalable multi-robot systems by partitioning the task domain. In: Towards Autonomous Robotic Systems. vol. 13546, pp. 282–292 (2022). https://doi.org/10.1007/978-3-031-15908-4_22
4. Feil-Seifer, D., Mataric, M.: Defining socially assistive robotics. In: Rehabilitation Robotics. pp. 465–468 (2005). https://doi.org/10.1109/ICORR.2005.1501143
5. Hamilton, J., Stefanakos, I., Calinescu, R., Cámara, J.: Towards adaptive planning of assistive-care robot tasks. In: Formal Methods for Autonomous Systems. vol. 371, pp. 175–183 (2022). https://doi.org/10.4204/EPTCS.371.12
6. Hebesberger, D., Körtner, T., Gisinger, C., Pripfl, J.: A long-term autonomous robot at a care hospital: A mixed methods study on social acceptance and experiences of staff and older adults. Int. J. Soc. Robotics **9**(3), 417–429 (2017). https://doi.org/10.1007/s12369-016-0391-6
7. Kephart, J.O., Chess, D.M.: The vision of autonomic computing. IEEE Computer **36**(1) (2003)
8. Lu, X., Woo, H., Faragasso, A., Yamashita, A., Asama, H.: Robot navigation in crowds via deep reinforcement learning with modeling of obstacle uni-action. Advanced Robotics **37**(4), 257–269 (2023). https://doi.org/10.1080/01691864.2022.2142068
9. Okamura, A.M., Matarić, M.J., Christensen, H.I.: Medical and health-care robotics. IEEE Robotics & Automation Magazine **17**(3), 26–37 (2010). https://doi.org/10.1109/MRA.2010.937861
10. Qureshi, M.A., Qureshi, K.N., Jeon, G., Piccialli, F.: Deep learning-based ambient assisted living for self-management of cardiovascular conditions. Neural Computing and Applications **34**(13), 10449–10467 (2022). https://doi.org/10.1007/s00521-020-05678-w
11. Reguii, I., Hassani, I., Rekik, C.: Mobile robot navigation using planning algorithm and sliding mode control in a cluttered environment. Journal of Robotics and Control **3**(2) (2022). https://doi.org/10.18196/jrc.v3i2.13765
12. Riley, J., Calinescu, R., Paterson, C., Kudenko, D., Banks, A.: Reinforcement learning with quantitative verification for assured multi-agent policies. In: Agents and Artificial Intelligence. pp. 237–245 (2021). https://doi.org/10.5220/0010258102370245
13. Silva Mendoza, S., Verdezoto, N., Paillacho, D., Millan Norman, S., Hernandez, J.D.: Online social robot navigation in indoor, large and crowded environments. In: Robotics and Automation (2023), https://orca.cardiff.ac.uk/id/eprint/156489/
14. Stefanakos, I., Calinescu, R., Douthwaite, J.A., Aitken, J.M., Law, J.: Safety controller synthesis for a mobile manufacturing cobot. In: Software Engineering and Formal Methods. vol. 13550, pp. 271–287 (2022). https://doi.org/10.1007/978-3-031-17108-6_17

15. Vázquez, G., Calinescu, R., Cámara, J.: Scheduling of missions with constrained tasks for heterogeneous robot systems. In: Formal Methods for Autonomous Systems. vol. 371, pp. 156–174 (2022). https://doi.org/10.4204/EPTCS.371.11

Evaluation of Safety Constraints in Autonomous Navigation with Deep Reinforcement Learning

Brian Angulo[1], Gregory Gorbov[1,3], Aleksandr Panov[2,3], and Konstantin Yakovlev[2,3]

[1] Moscow Institute of Physics and Technology, Dolgoprudny, Russia
[2] AIRI, Moscow, Russia
[3] Federal Research Center for Computer Science and Control of Russian Academy of Sciences, Moscow, Russia

Abstract. While reinforcement learning algorithms have had great success in the field of autonomous navigation, they cannot be straightforwardly applied to the real autonomous systems without considering the safety constraints. The later are crucial to avoid unsafe behaviors of the autonomous vehicle on the road. To highlight the importance of these constraints, in this study, we compare two learnable navigation policies: safe and unsafe. The safe policy takes the constraints into account, while the other does not. We show that the safe policy is able to generate trajectories with more clearance (distance to the obstacles) and makes less collisions while training without sacrificing the overall performance.

Keywords: Autonomous Navigation · Reinforcement Learning · Safety Constraints.

1 Introduction

Deep Reinforcement Learning has demonstrated tremendous success in many high-dimensional control problems, including autonomous navigation. Within RL, the interaction of the agent with the environment is modeled as a Markov decision process (MDP) [1], where the goal is to optimize the expected cumulative reward. The agent in MDP has a big freedom to explore any behavior which could improve its performance, including those that might cause damage. To this end, it is crucial to ensure safety constraints. A well-known approach to consider safety constraints in RL is a Constrained Markov Decision Process (CMDP) [2]. A survey of methods for solving CMDP can be found in [3]. In this short paper, we will apply safety constraints to ensure the safe behavior of the autonomous vehicle. Particularly, we will use the Lagrangian method [4] that is one of the most widely used approaches for solving CMDP.

2 Problem Statement

We are interested in algorithms for autonomous navigation which provide certain safety guarantees. To this end, we model our problem as a CMDP, where the

agent (autonomous vehicle) must generate a sequence of actions (trajectory) that drives it to a goal while avoiding obstacles and ensuring a tolerable safety cost limit. The latter in our work is interpreted as an upper limit of the vehicle's velocity when it is moving near to the obstacles.

Trajectory generation We are interested in autonomous vehicles whose dynamics is described as [5]: $\dot{x} = vcos(\theta), \dot{y} = vsin(\theta), \dot{\theta} = \frac{v}{L}\tan(\gamma)$, where x,y are the coordinates of the vehicle's rear axle, θ is the orientation, L is the wheel-base, v is the linear velocity, γ is the steering angle. The former three variables comprise the state vector: $\boldsymbol{x}(t) = (x, y, \theta)$. The latter two variables form the control vector: $\boldsymbol{u}(t) = (v, \gamma)$, which can also be re-written using the acceleration a and the rotation rate ω as follows: $v = v_0 + a \cdot t, \gamma = \gamma_0 + \omega \cdot t$.

The robot is operating in the 2D workspace populated with static obstacles. Their shapes are rectangular (as the one of the robot). Let Obs denote the set of obstacles. Denote by $\mathcal{X}_{free}$ all the configurations of the robot which are not in collision with the obstacles. The problem is to find the actions that move the vehicle from its start configuration s_{start} to the goal one s_{goal}, s.t. that the kinodynamic constraints are met and the resultant trajectory lies in $\mathcal{X}_{free}$. These controls are generated using a sequential decision making based on CMDP.

Constrained Markov Decision Process Formally, CMDP can be represented as a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \mathcal{C}, d, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P}$ is the state-transition model, $\mathcal{R}$ is the reward function, $\mathcal{C}$ is a constraint cost function and γ is the discounting factor. During learning at each time step the agent being in a state $s_t \in \mathcal{S}$ takes an action $a_t \in \mathcal{A}$ and receives a reward $r_t \in \mathcal{R}$ and a cost $c_t \in \mathcal{C}$. The goal is to learn a policy, i.e. the mapping from the states to the distributions of actions, $\pi : \mathcal{S} \to P(\mathcal{A})$. The policy should maximize the expected return $J(\pi)$ from the start state s_t while satisfying the discounted cost with tolerable limit d_i trough the discounted cost $C(\pi)$ under policy π:

$$J(\pi) = \mathbb{E}_{\tau\sim\pi}[\sum_{i=0}^{T} \gamma^t r(s_t, a_t, s_{t+1})], \quad C(\pi) = \mathbb{E}_{\tau\sim\pi}[\sum_{t=0}^{T} \gamma^t c_i(s_t, a_t, s_{t+1})],$$

where $\tau = (s_0, a_0, s_1, a_1, \ldots)$ denotes a trajectory. The objective of CMDP for the policy π is to find: $\pi^* = \arg\max_{\pi\in\Pi} J(\pi), \quad s.t.\ C_i(\pi) \leq d_i$.

3 Method

To evaluate the influence of safety constraints, we will use a policy-gradient algorithm PPO [6] and its safe version – LagrangianPPO (LPPO) [4]. The learning and evaluation of these two algorithms are conducted in a gym environment from [7]. Next, we will briefly introduce some details of environment.

Environment Our environment provides an autonomous vehicle equipped with kinematic bicycle model and lidar-sensor in an environment which is populated with static obstacles as shown in Fig. 1. The vehicle state is described as a tuple $\boldsymbol{x} = (x, y, \theta, v, \gamma)$. We consider agent's state $s_t = (\Delta x, \Delta y, \Delta\theta, \Delta v, \Delta\gamma, \theta, v, \gamma, \boldsymbol{l})$ where Δx_j is the difference between the j-element of the tuple between goal and

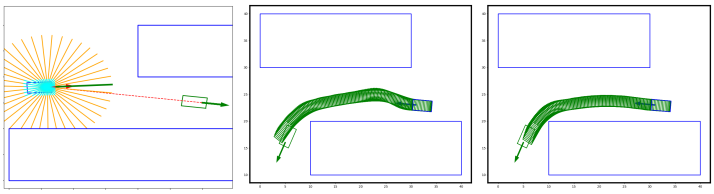


Fig. 1. The learning environment is shown **in the left figure**. **Blue** rectangle with **Red** arrow is the current state of the robot. **Green** rectangle is the goal desired state. **Blue** rectangles are the static obstacles. **Orange** lines are the lidar rays and **Cyan** lines are the safety radius. **The center and right figures** represent the trajectories generated by LPPO (safe) and PPO (unsafe) policy for the same task respectively.

current vehicle state, θ, v, γ are elements of current vehicle state and $\boldsymbol{l}$ is a tuple of rays-measurements from lidar. In this paper, we consider the same actions and reward function from [7]. The actions $a_t = (a, \omega)$ are composed of the linear acceleration a and rotation rate ω. For details of the actions, kinematic model and reward function we refer the readers to [7].

Safety Constraints The safety constraint cost is induced by a velocity of the autonomous vehicle at which it moves near to the obstacles due to the risk of collision like the method proposed in [8]. We impose an immediate constraint cost as $c_i(s_i) = \|v_i\| \times 1\{\|\boldsymbol{l}\| \leq r_{safety}\}$, where $r_{safety} = 0.5m$ is the safety radius – see Fig. 1, $\boldsymbol{l}$ is the current lidar signal, and 1 is boolean function. This safety constraint allows to the agent to move near obstacles at tolerable velocity v_i so as not to violate the tolerable limit d_i.

4 Evaluation

We trained two policies, i.e. PPO [6] and LPPO [4] on a dataset from [7]. The dataset is comprised of tasks which consist of start, goal and a set of static obstacles. The goal is to generate a trajectory from start to goal. In the Fig. 2 we show the learning process of these algorithms. We can see that PPO starts to converge earlier in comparison with LPPO. But at the same time it violates more safety constraints. On the other hand, the LPPO agent tries to find a trade-off.

After training we evaluate these two policies on a validation dataset through the following metrics: success rate (SR, %), collision rate (CR, %) and mean minimum clearance distance (MMC, m) of the successfully generated trajectories. The results for PPO are $SR = 96.25\%$, $CR = 3.75\%$, $MMC = 0.61m$, and for LPPO are $SR = 94.5\%$, $CR = 3.5\%$, $MMC = 0.67m$. We see that the LPPO agent tends to avoid the obstacles with more clearance in order to guarantee the safety constraints as shown in the Fig. 1. On the other hand, the PPO agent acquired more aggressive behavior that tries to reach the goal as soon as possible regardless of the risk of collision. Additionally, we can see that LPPO agent finishes some tasks without success or collision. In this cases, the LPPO

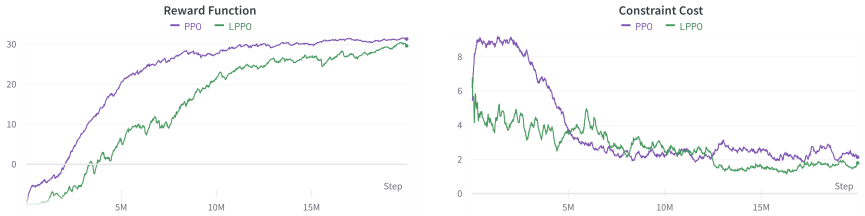


Fig. 2. A comparison of learning curves between PPO and LPPO.

agent prefers to make an abrupt stop instead of colliding with an obstacle. The latter is a safe behavior that agent acquired thanks to the safety constraints.

5 Conclusion

In this short paper, we have evaluated the consideration of safety constraints during the optimization process for policy gradient algorithm PPO. While PPO can converge earlier and has a slightly high success rate, the learned behavior of LPPO is more conservative due to taking the safety constraints into account. Overall, considering safety constraints in the optimization problem provides a more conservative and safe behavior without sacrificing of performance.

References

1. R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
2. E. Altman, *Constrained Markov decision processes*. CRC press, 1999, vol. 7.
3. Y. Liu, A. Halev, and X. Liu, "Policy learning with constraints in model-free reinforcement learning: A survey," in *The 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 2021.
4. Y. Chow, M. Ghavamzadeh, L. Janson, and M. Pavone, "Risk-constrained reinforcement learning with percentile risk criteria," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6070–6120, 2017.
5. B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.
6. J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *CoRR*, vol. abs/1707.06347, 2017. [Online]. Available: http://arxiv.org/abs/1707.06347
7. B. Angulo, A. Panov, and K. Yakovlev, "Policy optimization to learn adaptive motion primitives in path planning with dynamic obstacles," *IEEE Robotics and Automation Letters*, 2022.
8. Y. Chow, O. Nachum, A. Faust, E. Duenez-Guzman, and M. Ghavamzadeh, "Lyapunov-based safe policy optimization for continuous control," *arXiv preprint arXiv:1901.10031*, 2019.

Human evaluation of robotic grippers for berry picking

Laura Álvarez-Hidalgo & Ian S. Howard

SECAM, University of Plymouth, PL4 8AA, Plymouth
laura.alvarezhidalgo@students.plymouth.ac.uk, ian.howard@plymouth.ac.uk

Abstract. We describe the construction and evaluation of two robotic grippers for berry picking. Using a pneumatic cylinder drive, one was constructed from hard materials and the other from soft materials. A novel evaluation paradigm using a handle mechanism was developed, so the grippers could be directly operated by human participants. An artificial bush was also constructed and used for evaluation purposes. Overall, both grippers performed worse than the human hand, indicating that further development is needed.

Keywords: Robotics, Human Evaluation, Remote Operation, Agriculture, Soft Robotics, Grippers, Raspberry.

1 Introduction

There is a growing interest in robotic harvesting technologies [1, 2]. Recently there has also been significant development in the field of 3D printed robot components for harvesting, including the use of soft grippers [3], which are an area of intense research [4] and offer benefits such as low cost and ease of manufacture [5]. Here we investigate soft and hard grippers for integration in low-cost agricultural robot systems. We use human evaluation using the grippers as tools in picking tasks, thereby decoupling the operation of potential AI controllers from the intrinsic capabilities of each gripper design.

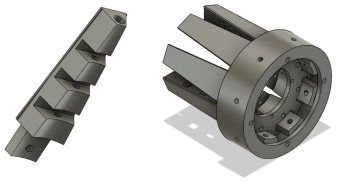


Fig. 1. Soft Gripper Finger and Hand.

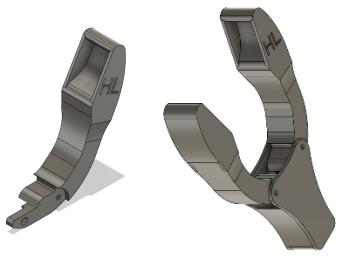


Fig. 2. Hard Gripper Finger and Hand.

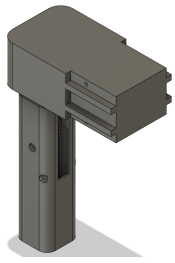


Fig. 3. Handle.

2 Mechanism Design

We first developed a soft gripper with eight Ninja flex fingers, which can flex and grasp objects when its tendons are tensioned. To allow the fingers to bend, each consisted of a continuous back structure with four protruding block sections, separated by gaps (see Fig. 1 Finger). A string tendon was inserted

through a small hole in each finger block all the way up to the fingertip, where it was firmly secured by a knot in the end of the string and glued to secure it in place. Pulling the tendon closed the gaps between the blocks and caused the finger to bend. A small location hole at the lower end of each finger enables the attachment of all eight fingers to a circular hand structure using small screws (Fig. 1. Hand). A circular plate was fastened to the end of the pneumatic cylinder and all tendons were attached to it, and pulling the plate closed all eight fingers simultaneously. See Fig. 4A to see the principle of operation. When tendon tension was released, the springiness of the Ninja Flex allowed all fingers to return to their original resting position without the need for active retraction.

The second gripper design used hard materials (Fig. 2). It comprises a pair of PLA+ fingers that were driven open and closed by an air cylinder acting on the end of a parallelogram mechanism (the principle of operation shown in Fig. 5A). To ensure the gripper could effectively grasp berries, both its fingers had a curved opening at their tips designed to accommodate an average sized berry.

To test the grippers, a PLA+ handle was developed (Fig. 3), which enabled human participants to operate them as tools in picking tasks (Figs. 4B & 5B). The handle has a diameter of 33mm to ensure that it could comfortably fit a human operator's hand. An internal microswitch was housed within the handle and used to activate a pneumatic actuator. The handle was hollow and the control cables from the switch to the Arduino Nano exited through its base. Its top provided a gripper attachment point, to ensure interchange between them was quick and easy.

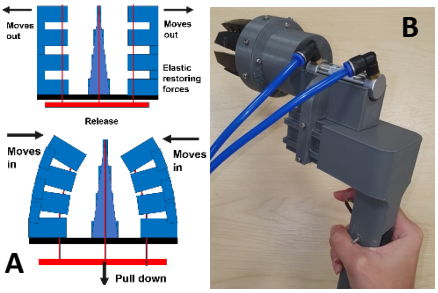


Fig. 4. Soft Gripper. **Panel A** Principle of Operation. **Panel B** Gripper Attached to Handle.

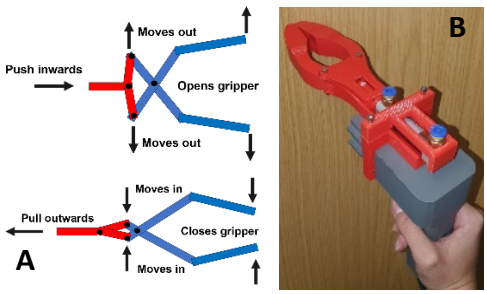


Fig. 5. Hard Gripper. **Panel A** Principle of Operation. **Panel B** Gripper Attached to Handle.

A Heschen CDJ2B 16-25 slim, lightweight (20g) double-acting pneumatic cylinders was chosen to actively close the gripper mechanisms (and also open the hard gripper), since it provides a clean and compact form of actuation

[6]. The cylinder had a 16mm bore and 25mm stroke. Running at 5 bar air pressure it could generate up to 50N force. It was controlled using a Heschen 4V210 5-way 2-position 24-volt solenoid valve, which enabled the cylinder rod to be actively driven in and out. Quickfit 6mm ports and 6mm o/d polyethylene tubes were used to connect the air supply, which was provided by a small Clarke air Shhh 50/24 silent air compressor. Both grippers were operated using a switch on the handle mechanism that activated the solenoid valve using a driver circuit implemented on an Arduino Nano (Fig. 6).

To evaluate the grippers' ability to pick berries, a test bush was constructed from a wooden rack (Fig. 7A). Artificial raspberries were hung using Ninja Flex stalks, which released the berry when pulled with sufficient force. This provided a controlled simulation of real berry picking. To achieve this, the berry stem had a cone-like shape that fitted into holes at the base of the berry. The stems themselves were attached to the rack using a simple screw fastening mechanism, allowing for appropriate placement to mimic their natural growth See Fig. 7B to see the 3D printed berry stems.

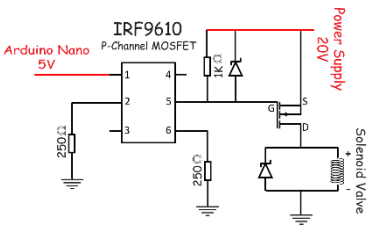


Fig. 6. Circuit for Pneumatic Controller

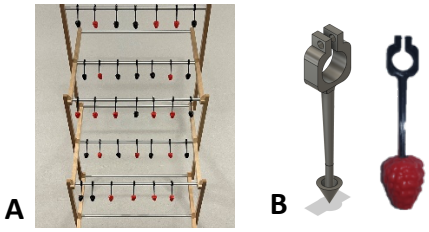


Fig. 7. Test Berries and Bush

3 Results

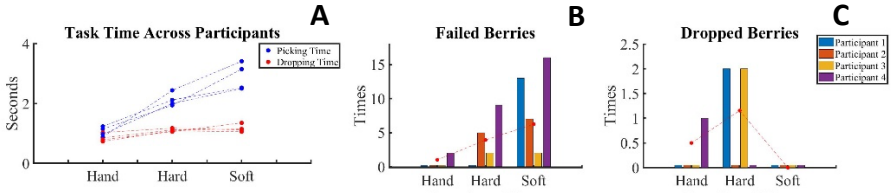


Fig. 8. Experimental Results for each Gripper Mechanism. Panel A: Average Picking and Release Time across Participants. Panel B: Berries Missed Panel C: Berries Dropped.

Four participants performed timed experiments to evaluate the effectiveness of using a hard and soft gripper, as well as their hands, in a picking task on an artificial berry bush. Participants were required to pull-off 34 artificial raspberries and collect them in a bowl. A video link showing the picking task is provided **here:**
https://youtube.com/playlist?list=PLFlgfzQylyK69KlPiWzTve7EWyr2Um5gP

We found the time taken to pick a berry varied substantially between conditions, as shown in the plots of the results in Fig. 8. The hand took an average of 1.0s to pick a berry, whereas it took 2.1s and 2.9s with the hard and soft grippers, respectively. Although the soft gripper took almost twice the time compared to the hard gripper to pick the berry, the time advantage was reduced considerably when berry dropping was considered. Results also showed that using the soft gripper, participants tended to miss the berry more often, but once picked, it had a lower chance of being dropped. Conversely, using the hard gripper, participants tended to be more successful in picking a berry but dropped it more often during transfer to the bowl.

4 Discussion

Using human evaluation, we demonstrated that two robotic grippers could successfully harvest berries from a test raspberry bush. A hard gripper performed better in terms of berry grasping, whereas a soft gripper exhibited fewer berry dropping errors. Both were worse than using the human hand directly. Clearly, the gripper designs need to be improved, and future experiments are needed to further compare harvesting performance. We suggest that tracking and video of the picking task performed by human operation may provide a useful dataset to train future autonomous robotic systems. Finally, we thank the University of Plymouth for Proof-of-Concept support.

5 References

[1] F. Rodríguez, J. Moreno, J. Sánchez, and M. Berenguel, "Grasping in agriculture: State-of-the-art and main characteristics," *Grasping in robotics,* pp. 385-409, 2013.

[2] E. Navas, R. Fernández, D. Sepúlveda, M. Armada, and P. Gonzalez-de-Santos, "Soft gripper for robotic harvesting in precision agriculture applications," in *2021 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2021: IEEE, pp. 167-172.

[3] E. Navas, R. Fernandez, D. Sepúlveda, M. Armada, and P. Gonzalez-de-Santos, "Soft grippers for automatic crop harvesting: A review," *Sensors,* vol. 21, no. 8, p. 2689, 2021.

[4] J. Hughes, U. Culha, F. Giardina, F. Guenther, A. Rosendo, and F. Iida, "Soft manipulators and grippers: a review," *Frontiers in Robotics and AI,* vol. 3, p. 69, 2016.

[5] F. Kurbah, S. Marwein, T. Marngar, and B. K. Sarkar, "Design and development of the pineapple harvesting robotic gripper," *Communication and Control for Robotic Systems,* pp. 437-454, 2022.

[6] H. I. Ali, S. Noor, S. Bashi, and M. H. Marhaban, "A review of pneumatic actuators (modeling and control)," *Australian Journal of Basic and Applied Sciences,* vol. 3, no. 2, pp. 440-454, 2009.

Toward Semi-Autonomous Terrestrial Robots for Atmospheric Electricity Measurement

Gjosse Zijlstra, Karen L. Aplin, and Edmund R. Hunt

University of Bristol, Bristol BS8 1QU, UK
{gjosse.zijlstra,karen.aplin,edmund.hunt}@bristol.ac.uk

Abstract. We explore the use of a rover robot carrying an electric field mill for ground-based atmospheric electricity measurement. The robot platform includes a 2D LiDAR and stereo image camera for terrain mapping, with the aim of generating new insights into atmospheric electricity processes. Initial teleoperated experiments allowed for the plotting of measured voltages on a map using GNSS. We plan to explore the possibility of cooperative 3D mapping behaviours in a 'sparse swarm' configuration of around six robots, with expert-in-the-loop oversight. Ultimately, we intend for this research to inform opportunities in atmospheric electricity research on other planets, such as Mars.

Keywords: Electrostatics · Terrain Mapping · Sparse Swarms

1 Introduction

Atmospheric electricity is still poorly understood, on Earth and other planets [2]. Developments in robotics technology, especially in Unmanned Aerial Vehicles (UAVs), are leading to increased interest in atmospheric electricity research opportunities with aerial robotic platforms (e.g. [4], [7]). Relatively little work has been done with terrestrial robots (see [1] as a first example). Here, we present initial findings from experiments using one rover robot carrying an electric field mill for ground-based atmospheric electricity measurement. Beyond this initial phase of the research project, ultimately we aim to deploy a 'sparse swarm' [8] of multiple rovers dispersed across a wide geographic area, operating with some level of autonomy to propose their movement through the environment with expert input and oversight. This should generate new insights into atmospheric electricity processes: for instance tracking of dust devils and storms on Mars, or comparing electrostatic weather phenomena between Earth and Mars.

2 Robotic Platform

The use of robot platforms for scientific survey presents several potential advantages [3], including the efficient and precise measurement of physical processes across wide areas, especially if multi-robot systems can be deployed.

Our base robotic platform is a 'Leo Rover': a 4-wheeled rover controlled by a Raspberry Pi. Designed as a Mars-type rover, it has a linked rocker suspension, making it well-suited to outdoor exploration (Figure 1). On top of the rover, we add a sensor payload, including: a 2D LiDAR (RPLIDAR A1); stereo image camera (Intel RealSense D455); a GNSS receiver (global navigation satellite system) with RTK (real-time kinematic positioning) capabilities; and an upward-facing electric field mill (JCI-140). To provide extra computing power for the above-mentioned sensors a LattePanda Delta (a single-board computer with an integrated Arduino) is connected to the Raspberry Pi controller over Ethernet.

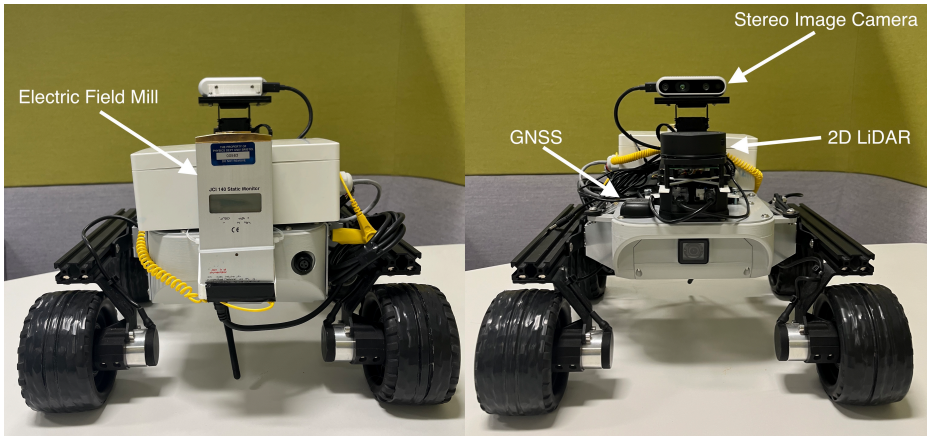


Fig. 1: Leo Rover (back and front) with labelled sensors

The Raspberry Pi operates as the ROS (Robotic Operating System) master, connected to the LattePanda running sensor processing nodes. The Pi is used to do all the physical control of the robot, and it also publishes nodes for both the IMU (inertial measurement unit) and wheel encoders. The LattePanda handles the more computationally intensive tasks of combining the stereo image camera and LiDAR to do Visual SLAM(Simultaneous localization and mapping). The ROS package RTAB-Map [6] is used for this task and allows the creation of a detailed 3D terrain map as the robot moves, which along with the readings from the electric field, will allow the generation of a composite electrostatic map of the outdoor environment [5]. With the deployment of multiple robots (e.g., up to six across an area of multiple kilometres square), this map can be created more rapidly, and also give novel insight into phenomena in atmospheric electricity, for instance in how it interacts with the local terrain.

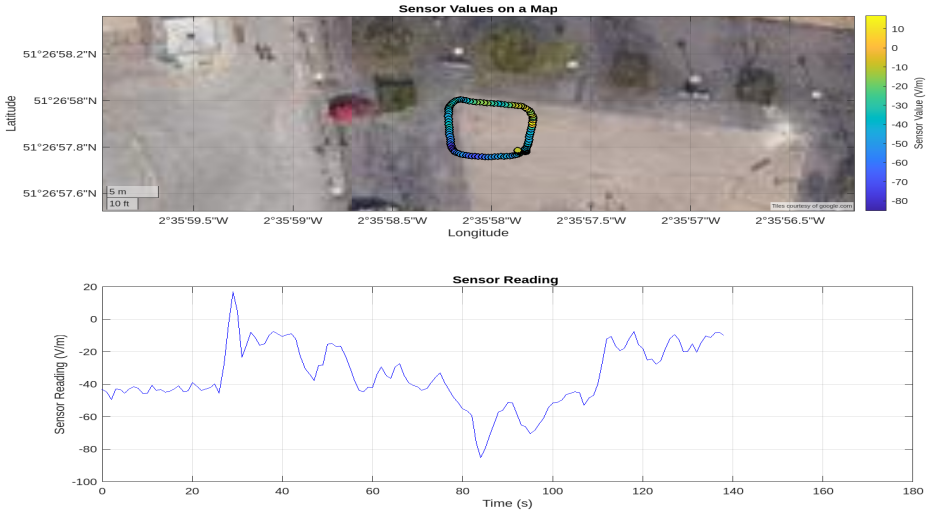


Fig. 2: Trial 1, Date: 21/07/2023, Start time: 09:24 AM UTC, Scattered clouds

Fig. 3: Trial 2, Date: 21/07/2023, Start time: 09:30 AM UTC, Scattered clouds

3 Initial Trials

Initial teleoperated experiments across a relatively small area (around 20 m^2) used one robot and allowed the plotting of measured voltages on a geographic map using GNSS. Two trials were conducted in the same location, an open urban space. The results of the trials can be seen in Figure 2 and Figure 3, whereby the dots show the electric field measured at each GNSS location, and the line graph shows the electric field meter reading over time. The robot travels in a

counterclockwise loop. Plausible voltage readings were obtained in the range -100 to 15 V. A good comparison source is Reading Atmospheric Observatory station[1] which provides daily meteorological data. As we measure the electric field and not the potential gradient a negative value is expected in fair weather.

4 Future Plans

There are several planned work packages to further this research. First, we will calibrate the electrostatic sensor, using a setup where the robot is placed between two metal plates applied with a set known voltage, to determine the relevant correction. Second, we will use *Ansys Maxwell* software electromagnetic field modelling to obtain an appropriate geometric field enhancement factor for the robot, to crosscheck the experimental correction. Third, we will deploy multiple robots with an intermittent communication architecture to create a sparse swarm [8] of electrostatic robots. We will demonstrate cooperative swarm behaviour using (semi-)autonomous choices on where to focus mapping, including the use of expert user input. Fourth, we will develop a simulation of robot swarm electrostatic field measurement on Mars, with a view to the long-term potential of this research in space missions.

References

1. Aplin, K., Xiong, Z.: Electric field measurements made on a robotic platform. In: *Proceedings 2021 Annual Meeting of the Electrostatics Society of America*, 14-16 June (2021). https://doi.org/10.48550/arXiv.2106.00407
2. Aplin, K. L.: Electrifying atmospheres: charging, ionisation and lightning in the solar system and beyond. SpringerBriefs in Astronomy, Springer: New York (2013). https://doi.org/10.1007/978-94-007-6633-4
3. Dunbabin, M,. Marques, L.: Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics & Automation Magazine* 19.1 (2012): 24-39. https://doi.org/10.1109/MRA.2011.2181683
4. Harrison, R. G., et al.: Ionic charge emission into fog from a remotely piloted aircraft. *Geophysical Research Letters*, 49(19), e2022GL099827 (2022). https://doi.org/10.1029/2022GL099827
5. Katz R., et al.: Integrated Sensing Framework for 3D Mapping in Outdoor Navigation, in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 2264–2269. https://doi.org/10.1109/IROS.2006.282571
6. Labbé, M., Michaud F.: RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of Field Robotics* 36.2 (2019): 416-446. https://doi.org/10.1002/rob.21831
7. Schön, M., et al.: Fair-Weather Atmospheric Charge Measurements with a Small UAS. *Journal of Atmospheric and Oceanic Technology*, 39(11), 1799-1813 (2022). https://doi.org/10.1175/JTECH-D-22-0026.1
8. Tarapore, D., et al.: Sparse robot swarms: Moving swarms to real-world applications. *Frontiers in Robotics and AI*, 7, 83 (2020). https://doi.org/10.3389/frobt.2020.00083

[1] https://research.reading.ac.uk/meteorology/atmospheric-observatory/

Reinforcement Learning-based Adaptive Probabilistic Movement Primitives in Hybrid Scenarios (Extended Abstract)

Shibao Yang, Pengcheng Liu, and Nick Pears

University of York, York, UK

1 Introduction

Robotic manipulation has been a subject of extensive research, with a primary focus on static or dynamic environments. However, real-world scenarios often present a blend of both, termed as 'hybrid scenarios', where static and movable objects coexist. These hybrid environments pose unique challenges, including the need to distinguish between static and movable objects, adapt to unpredictable shifts, and respond to the activities of other agents [4, 1]. Despite the prevalence of such scenarios, they have been largely overlooked in the current body of research.

The complexity of hybrid scenarios is further amplified when we consider the need for a robotic arm to adapt to a wide array of situations that may arise within these environments. These situations could range from interactions with static and movable objects to a combination of both within a single scene. The ability to navigate these challenges effectively is crucial for the successful deployment of robotic systems in real-world settings [2].

In response to this gap in the field, we propose an adaptive probabilistic motion primitive (ProMP) [6, 2] based approach. This approach combines non-prehensile actions [5] and vision techniques [7] to enhance the learning of the properties of the hybrid scene. This enables the robot arm to accurately identify object states and generate appropriate trajectories for the manipulation task. The main components of our approach are an exploration module and an adaptive ProMPs framework for trajectory generation.

Our research aims to extend the current understanding and capabilities of robotic manipulation in hybrid scenarios. By developing a reinforcement learning-based pipeline that includes exploration, object state detection, and adaptive trajectory generation, we seek to improve the accuracy, efficiency, and adaptability of robot arm motion planning and manipulation. This method has the potential to significantly enhance the performance of robotic systems in real-world applications, where environments are often dynamic and unpredictable.

The primary objectives of the research are to:

- Develop an exploration module to efficiently explore and understand hybrid scenarios. This should accurately perceive the environment and build

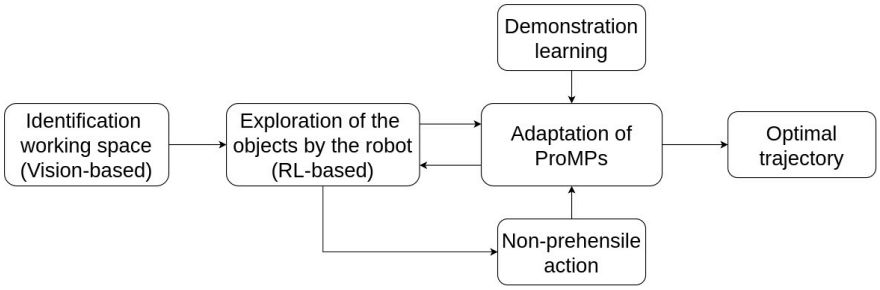


Fig. 1: RL-based Adaptive ProMPs.

a representation that captures the location, properties, and types of objects (static and movable).
– Extend the existing ProMPs framework to autonomously adapt to the problems of motion planning in hybrid scenarios, such as moveable obstacles interfering with predicted motion trajectories.
– Evaluate the performance of the proposed pipeline through simulations and real-world experiments, focusing on accuracy, robustness, and adaptability.

1.1 Method

The Fig. 1 shows the structure of the RL-based adaptive ProMPs.

– **Vision Technique**: A fully convolutional network (FCN) [9] can be used to facilitate a more nuanced segmentation process. This implementation permits the robotic system to achieve a more comprehensive understanding of the spatial relationships between different objects. When integrated with the Robot Operating System (ROS) within the Gazebo simulator, the resulting system is able to perform an accurate object segmentation within the environment.
– **Non-prehensile Actions**: In the exploration module, we utilize reinforcement learning (RL) algorithms, specifically Proximal Policy Optimization (PPO) [8], to facilitate the discovery of feasible trajectories within hybrid-scenario environments by the Franka robot arm. The PPO algorithm has been chosen due to its recognized sample efficiency, stability, and reliability - attributes that render it suitable for the complexities of hybrid environments. Furthermore, PPO's robustness to hyperparameter choices, ease of implementation, compatibility with prevalent RL libraries, as well as its scalability for large neural networks and parallelized training, make it well-suited for tasks involving high-dimensional state and action spaces that characterize hybrid scenarios.
– **Adaptive ProMPs for Trajectory Generation**: We extend the existing Probabilistic Movement Primitives (ProMPs) framework [3] to handle hybrid scenarios. To do this, we incorporate the property of the scenario

into the ProMPs model. We propose adding virtual via-points around obstacles, calculated as the closest points on the obstacle's surface moved away in the direction of the normal vector. Conditioning the ProMP distribution on these via-points allows the generated trajectory to circumvent different types of obstacles, thereby optimizing the trajectory.

In our study, we developed a hybrid-scenario experimental framework involving four types of items: static boxes, spheres, cylinders, and movable boxes. These items were arranged into three distinct settings, each increasing in complexity.

The experimental process began with the robotic arm identifying and navigating obstacles using a combination of reinforcement learning and sensory inputs, including vision. We employed a Fully Convolutional Network (FCN) to process scene images and deduce the 3D location of each object, with obstacles marked in distinct colors for clarity.

We incorporated Learning from Demonstration (LfD) to train Probabilistic Movement Primitives (ProMPs), which were then updated or "conditioned" to reflect the current state of the environment. In hybrid scenarios, we designated via-points around obstacles to condition the ProMPs for adaptation, ensuring trajectory generation that was attuned to the current environment.

When a movable obstacle appeared in the predicted trajectory, the robot arm executed a non-prehensile action to displace the obstacle, maintaining the fluidity of the process. Finally, an optimal trajectory was sampled from the conditioned ProMPs, representing a smooth and efficient movement of the robot arm that considered the current environment and any remaining obstacles.

1.2 Evaluation

We will evaluate the proposed pipeline through a series of simulation-based and real-world experiments. The performance metrics will include:

- **Accuracy**: The ability of the pipeline to generate appropriate trajectories based on the detected object states.
- **Efficiency**: The time required for the adaptive ProMPs to generate and execute trajectories.
- **Adaptability**: The capability of the pipeline to handle changes in the environment and object states, demonstrating its robustness in dynamic situations.

Accuracy verification starts with the success rate of the motion planning, the ability to successfully avoid dynamic objects and the ability to ensure safe movement of the robot arm.

We record the time taken by the adaptive ProMPs to generate a trajectory after conditioning on the current environment. We perform this measurement across multiple trials and compute the average trajectory generation time. This will help assess the computational efficiency of the ProMPs. In addition to evaluating efficiency in terms of time, consider other performance metrics such as

trajectory smoothness, task success rate, and safety (e.g., collision avoidance). This will provide a more comprehensive view of the adaptive ProMPs' performance in the hybrid-scenario environment.

We will evaluate the ability of ProMPs to adapt to external perturbations, such as sudden changes in the environment or the presence of obstacles. Design experiments where the learned ProMPs need to adapt in real-time to these perturbations and measure their performance in terms of task success rate, response time, and other relevant metrics. Compare the results with other methods to demonstrate the robustness of ProMPs in adapting to perturbations.

2 Conclusion

This research aims to extend the ProMPs framework for robotic manipulation in hybrid scenarios with static and movable objects. By developing a reinforcement learning-based pipeline that includes exploration, object state detection, and adaptive trajectory generation, we seek to improve the accuracy, efficiency, and adaptability of robot arm motion planning and manipulation. This method has the potential to significantly enhance the performance of robotic systems in real-world applications, where environments are often dynamic and unpredictable.

References

1. Dong, H., Asadi, E., Sun, G., Prasad, D.K., Chen, I.M.: Real-time robotic manipulation of cylindrical objects in dynamic scenarios through elliptic shape primitives. IEEE Transactions on Robotics **35**(1), 95–113 (2018)
2. Frank, F., Paraschos, A., van der Smagt, P., Cseke, B.: Constrained probabilistic movement primitives for robot trajectory adaptation. IEEE Transactions on Robotics **38**(4), 2276–2294 (2021)
3. Gomez-Gonzalez, S., Neumann, G., Schölkopf, B., Peters, J.: Adaptation and robust learning of probabilistic movement primitives. IEEE Transactions on Robotics **36**(2), 366–379 (2020)
4. Mohanan, M., Salgoankar, A.: A survey of robotic motion planning in dynamic environments. Robotics and Autonomous Systems **100**, 171–185 (2018)
5. Murali, P.K., Dutta, A., Gentner, M., Burdet, E., Dahiya, R., Kaboli, M.: Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter. IEEE Robotics and Automation Letters **7**(2), 4686–4693 (2022)
6. Paraschos, A., Daniel, C., Peters, J., Neumann, G.: Using probabilistic movement primitives in robotics. Autonomous Robots **42**(3), 529–551 (2018)
7. Riansyah, M.I., Nugraha, Y.P., Ridlwan, H.M., Trilaksono, B.R.: 3d mapping hexacopter simulation using gazebo and robot operating sytem (ros). In: Proceedings of the 9th International Conference on Machine Learning and Computing. pp. 507–510 (2017)
8. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
9. Zeng, A., Song, S., Welker, S., Lee, J., Rodriguez, A., Funkhouser, T.: Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4238–4245. IEEE (2018)

Suction Cup Detachment Mechanism based on Fluidic Soft Actuators Strain for In-vivo Applications

Saloni Hajare, Frederick Forbes, and Dana D. Damian

Sheffield Biomedical Robotics Lab, Automatic Control and Systems Engineering Department, University of Sheffield, Sheffield, S1 3PD, {sphajare1,fforbes2, d.damian}@sheffield.ac.uk

Abstract. Remotely controllable non-invasive surgical robots are an emerging technology in healthcare. Soft material fabricated suction cups can be used as a reliable means for adhering medical devices in-vivo. However, the safe detachment and removal of these medical devices can be challenging, often requiring the use of magnetic torque or a high-voltage electric field. This paper proposes a non-invasive method of detaching suction cups integrated into a soft device by using lateral strain from its elastomeric fluidic actuators. The study highlights initial findings and proposes design modifications to improve the soft device with future plans to adapt it for use in implants for regenerative medicine.

Keywords: Soft robotic implants · Elastomeric Fluidic actuators · Suction cups detachment

1 Introduction

Remotely controllable non-invasive surgical robots are an emerging technology in medicine. As part of this technology, the interaction at the interface between tissue and robots, by means of adherence is of interest. Although chemical glue, gecko-adhesion [1], electroadhesion [2] and magnetic field [3] are few means of adherence, the need for high voltage, bio-compatibility problems [3], makes their in-vivo use challenging. Passive suction cups, due to their minimal preload requirement, and absence of external force to keep them in position, are strong adhesives. They enhance the non-invasive nature of implants, eliminating the need for suturing. Despite significant research in enhancing adherence, the methods for safe detachment of adhesives have been limited to using high-voltage dielectric materials [4] or through external magnetic torque [3]. Inspired by use of elastomeric actuators for therapeutic purpose [5], this study proposes a new method of detaching suction cups using lateral strain from inflatable fluidic elastomeric actuators.

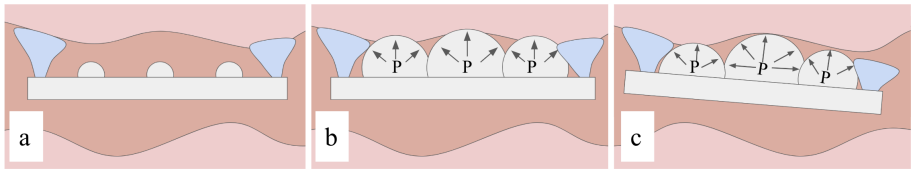


Fig. 1. a) Concept image of soft implant with suction cups b) Pnuematic ballooning actuators are inflated with pressure, P, to detach suction cup. c)Final detachment

2 Materials and Methods

2.1 Fabrication

The implantable soft robot consists of two suction cups and three ballooning membrane actuators as shown in Figure 1. Ecoflex™ 00-50 (Smooth On Inc.) was used for fabricating the implant due to its bio compliant material property. Cone shaped suction cups were used due to their higher adhesion strength on different surfaces [3].

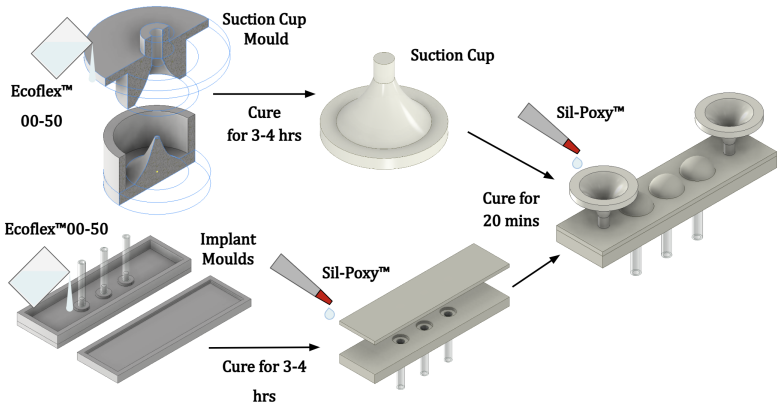


Fig. 2. Fabrication procedure

The fabrication steps are shown in Figure 2. For preparing the implant and suction cups, equal quantities of Ecoflex™ 00-50 part A and B were combined and mixed in an ARE-250 (Thinky) mixer for three minutes. The mixture was degassed and poured into resin moulds and left to cure for three to four hours at room temperature. The dimensions of the implant were 68x18x5 mm, with a 1mm thick ballooning membrane. The suction cups had a diameter of 18mm with 2 mm base thickness and were glued to the ends of the implant using Sil-Poxy™.

2.2 Experimental Procedure

Measurements for adhesion force of a single suction cup were performed in different environments - dry, moist, underwater and under oil. For each test the suction cup was placed on a Petri dish and a preload of 4.5 N was applied. The adhesion force data was measured by pulling the suction cup vertically using an IMADA tensile machine. The data was analysed in Matlab.

To investigate detachment, a smooth, wet glass slide was used as a substrate. The three actuators were manually controlled using 5ml syringes, and connected to a pressure sensor. The glass slide was pushed downwards, followed by a preload of 5N, which after trial and error was found to be the force required for proper adherence of the glass slide onto the suction cups. A pressure of 0.38 bar was used for inflation. Two inflation patterns were investigated: first in which the mid actuator, followed by outer actuators were inflated; in the second pattern, the outer actuators were inflated first, followed by the mid actuator.

3 Results and Discussion

Figure 3 shows the mean adhesion force of the suction cups and the limits of error bar show the standard deviation. In the moist and underwater conditions, the adhesion force was greatest at 2.96 and 2.65 N, respectively, followed by 1.79 N for dry and 0.95 N for oil. The higher efficacy of adhesion in underwater and moist conditions, makes the suction cups suitable for in-vivo use.

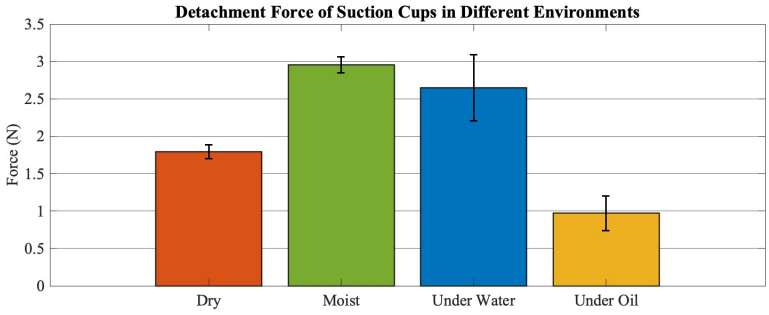


Fig. 3. Adhesion force of suction cups in different environments

The detachment experiment showed that by inflating all three actuators in any inflation pattern, the implant was able to overcome the adhesion force of a single suction cup. This is shown in Figure 4. Inflating the middle actuator applies a lateral strain on the outer actuators, resulting in a tangential force on the rim of suction cup, leading to detachment. The actuators can also be controlled to induce mechanostimulations for tissue regeneration [6][7]. By increasing the actuator diameter, and adding pressure control, failure due to overinflation can be minimised.

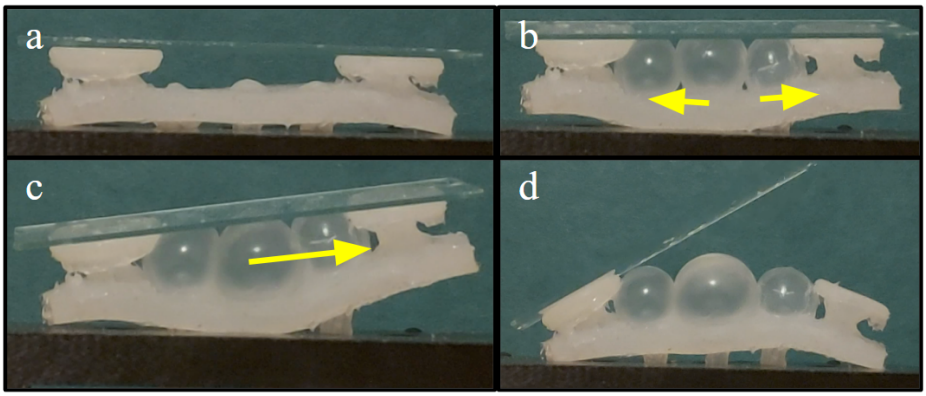


Fig. 4. Detachment cycle. The outer and mid actuators are inflated (b). Lateral strain between actuators leads to tangential force on a suction cup (c) causing its release (d)

4 Conclusion and Future Work

This work proposes the use of soft pneumatic actuators to achieve safe detachment of an implant. For future work, untethered means of inflation will be investigated to enhance safety and lifespan of the device for in-vivo use.

References

1. M. Zhou, N. Pesika, H. Zeng, Y. Tian, and J. Israelachvili, "Recent advances in gecko adhesion and friction mechanisms and development of gecko-inspired dry adhesive surfaces," *Friction*, vol. 1, no. 2, pp. 114–129, 2013.
2. K. Singh and S. Gupta, "Controlled actuation, adhesion, and stiffness in soft robots: A review," *J. Intell. Robotics Syst.*, vol. 106, no. 3, Nov 2022.
3. H. Iwasaki, F. Lefevre, D. D. Damian, E. Iwase, and S. Miyashita, "Autonomous and reversible adhesion using elastomeric suction cups for in-vivo medical treatments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2015–2022, 2020.
4. A. Zbiciak and T. Markiewicz, "A new extraordinary means of appeal in the polish criminal procedure: the basic principles of a fair trial and a complaint against a cassatory judgment," *Access to Justice in Eastern Europe*, vol. 6, no. 2, pp. 1–18, Mar. 2023.
5. H. Bakeri, K. Hasikin, N. A. Abd Razak, R. Mohd Razman, A. A. Khamis, M. nnuha, A. Tajuddin, and D. Reza, "Silicone elastomeric-based materials of soft pneumatic actuator for lower-limb rehabilitation: Finite element modelling and prototype experimental validation," *Applied Sciences*, vol. 13, no. 5, 2023.
6. D. D. Damian, K. Price, S. Arabagi, I. Berra, Z. Machaidze, S. Manjila, S. Shimada, A. Fabozzo, G. Arnal, D. V. Story, J. D. Goldsmith, A. T. Agoston, C. Kim, R. W. Jennings, P. D. Ngo, M. Manfredi, and P. E. Dupont, "In vivo tissue regeneration with robotic implants," *Science Robotics*, vol. 3, no. 14, p. eaaq0018, 2018.
7. E. Perez-Guagnelli, J. Jones, and D. D. Damian, "Hyperelastic membrane actuators: Analysis of toroidal and helical multifunctional configurations," *Cyborg and Bionic Systems*, vol. 2022, 2022.

Human Machine Interface for a Pick-and-Place Task Using Extended Reality Devices

Seth Roberts[1], Jonathan Eyre[2], and Dr Dana Damian[1][0000−0002−0595−0182]

[1] Automatic Control and Systems Engineering Department, University of Sheffield, Sheffield S1 3JD, United Kingdom
d.damian@sheffield.ac.uk
seth.roberts@hotmail.com
[2] Advanced Manufacturing Research Center, Sheffield S60 5TZ, United Kingdom

Abstract. Extended Reality (XR), including Augmented Reality (AR), technologies are being used within manufacturing to immerse operators into a 'cyber-physical' environment. This project investigates the use of a multimodal AR application to provide instructions over visual, auditory, and haptic modalities. A Microsoft HoloLens2 and two Bangle.JS wristwatches were chosen to deliver the experience. The trial task consisted of moving objects around a desk as a representative of a pick-and-place manufacturing activity. Statistical significance was found for the performance (time, errors) and experience with visual or auditory modalities compared to haptics. Overall, this study provided an insight into the utility of using visual, auditory, and haptic modalities within a simple manufacturing scenario.

Keywords: Extended Reality · Augmented Reality · Multi-modal Systems · Manufacturing · Pick and Place · HoloLens.

1 Introduction

Global manufacturing is currently undergoing a 'Fourth Industrial Revolution', marked by the blurring of the boundaries between the physical and digital worlds. Augmented Reality (AR) is uniquely positioned to lead this new technological era. AR is used to assist operators in their daily activities, with the most common applications being assembly, training, maintenance, and remote collaboration. This has enabled measurable benefits to the manufacturing process. For example, an experiment conducted by Z. H. Lai et al.[1] showed that using an AR system reduced the time and errors of assembly tasks.

AR in the manufacturing industry is promising but limited due to the lack of user-centred design approaches, such as natural interaction. A report by Aliprantis J et al. [2] found that user experience has not been properly considered when designing AR applications. R. K. Khamaisi et al.[3] found that AR applications developed using a user experience-centred design approach produced both objective and subjective benefits. Furthermore, multimodal user interfaces have been identified as a promising 3D user interaction technique. This paper compares the

performance of modalities in delivering a set of instructions as part of an AR application, where each instruction corresponds to a consistent user action.

2 Methodology

Manufacturing Use-Case The task chosen to mimic a manufacturing use-case was to move objects between four locations (zones) on a desk, all within a reachable distance. This was chosen as an abstract representation of a manufacturing task. However, it has direct relevance to order-picking tasks, such as sorting items on a conveyor belt, or re-configuring connections between manufacturing devices. The human users received one instruction at a time asking them to move an item between zones. The senses identified for feedback during the task were visual, auditory, and haptics. Each sense showed potential to be explored further within the manufacturing use-case.

AR Implementation The Microsoft HoloLens 2 was chosen as the primary AR device due to its availability. (Fig. 1 A). It ran an AR application built using the Unity Platform. Additionally, the Bangle.JS programmable watch was chosen to relay haptic cues due to its low price and simple programming interface (Fig. 1 A). The visual modality was implemented by placing an AR-rendered ring around a zone, while the auditory modality was implemented by playing a voice recording stating the name of a zone. The haptic instructions consisted of making the watches buzz in specific configurations. Fig. 1 B, C illustrates how these commands were provided to participants across each modality.

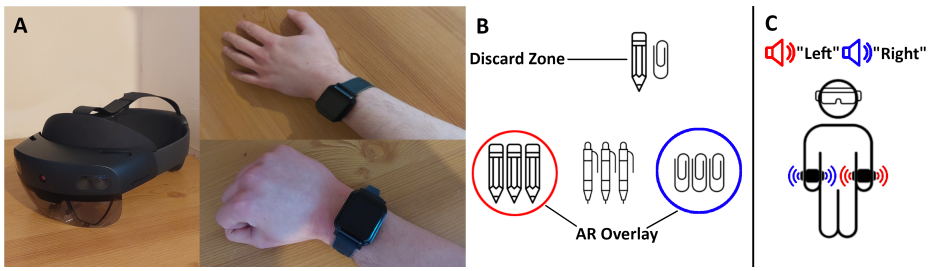


Fig. 1. A: Micosoft HoloLens2 (left) along with two Bangle.JS watches, one on the left hand, the other on the right hand B: The view of the user when the visual modality is enabled. Red: Pickup command, Blue: Put-down command. C: The auditory (a speaker playing a recording of the zone name) and haptic (a buzz on the wrist) modalities

Experimental Procedure The human user trials were ethically approved by the University of Sheffield. Ten participants, 2 females and 8 males, between

the ages of 22-30, mostly unaccustomed to using AR, took part in the experiment. The three modalities were trialled in isolation to gauge their effectiveness at instructing users to perform a simple task. The method used to indicate a zone over each modality is detailed in Table 1. Participants were informed on the nature of their task and the form of the instructions. Each trial consisted of 14 commands, 7 pickups and 7 placements. Participants' hands were tracked to automatically progress the sequence as they moved between zones. To adjust users to the new medium, they were allowed to have one practice trial using all the modalities simultaneously. The order of the modalities was randomised and uniformly distributed. Both quantitative (errors and completion times) and qualitative (usability survey) assessments were made to measure user satisfaction and performance for each modality. The time participants took to complete a task sequence was recorded, and mistakes that the users made were noted. Moreover, any participant comments, questions, or general frustrations were recorded. After the end of each task, they were asked to complete the Standard Usability Survey (SUS) [4], consisting of 10 questions answered on a Likert scale from 1-5.

Table 1. Cues for the zones in each modality.

Modality	Left Zone	Center Zone	Right Zone	Discard Zone
Visual	Coloured AR Ring Around Zone	Coloured AR Ring Around Zone	Coloured AR Ring Around Zone	Coloured AR Ring Around Zone
Auditory	"Left"	"Center"	"Right"	"Discard"
Haptics	A Left Wrist Buzz	A Buzz on both Wrists	A Right Wrist Buzz	Two Buzzes on both Wrists

3 Results

Participants took longer to complete tasks using the haptic modality than visual and auditory modalities; also making more mistakes, when performing a between-subjects comparison (Figure 2). Furthermore, participants completed tasks at a comparable rate when receiving instructions through audio over visuals, with only a marginal increase in mistakes. The spread of participant times, shown in Figure 2, details the variance in results for each modality. Using Welch's T-Test, statistical significance was found between the visual and auditory modalities against the haptic one, with powers of 0.58 and 0.53 respectively. Moreover, the results showed that participants enjoyed using the visual and auditory modalities over the haptics Table 2.

Table 2. Average SUS Scores for the modalities tested. A score of 68 or higher is considered better than average.

Modality	Visual	Auditory	Haptics
Average SUS Scores	78.25	76.5	54.2

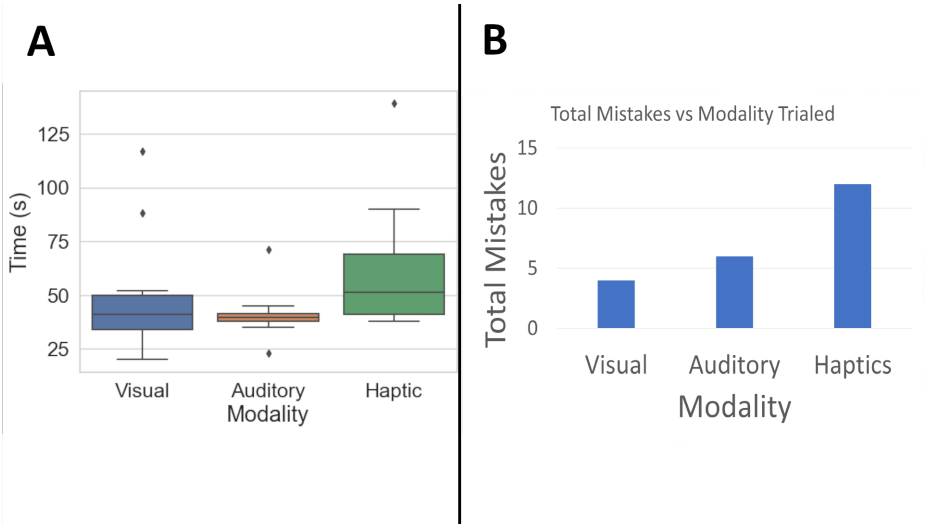


Fig. 2. Results of the trials. A: Box and Whisker Plot showing the timed results for each modality. The spread of completion times is shown along with various outlying pieces of data. B: The total number of mistakes made across all trials for each modality.

4 Conclusions and Future Work

A multimodal AR system was proven capable of delivering instructions to users in a simple manufacturing use-case. Trials showed that visual and auditory modalities performed better and had higher participant satisfaction than haptics. Further research should be done into how modalities may be used in combination and how each modality is best used in different manufacturing scenarios. This will improve operator satisfaction and effectiveness across manufacturing.

References

1. Z. H. Lai, W. Tao, M. C. Leu, and Z. Yin. Smart augmented reality instructional system for mechanical assembly towards worker-centered intelligent manufacturing. *Journal of Manufacturing Systems*, 55:69–81, 4 2020.
2. J. Izquierdo-Domenech, J. Linares-Pellicer, and J. Orta-Lopez. Towards achieving a high degree of situational awareness and multimodal interaction with ar and semantic ai in industrial applications. *Multimedia Tools and Applications*, 9 2022.
3. R. K. Khamaisi, E. Prati, M. Peruzzini, R.o Raffaeli, and M. Pellicciari. Ux in ar-supported industrial human–robot collaborative tasks: A systematic review. *Applied Sciences*, 11:10448, 11 2021.
4. J. Brooke. Sus: A quick and dirty usability scale. *Usability Eval. Ind.*, 189, 4 1995.

Access Control Mechanism Framework for Assistive Robots in Healthcare

Kavyan Zoughalian[1], Jims Marchang[2,3], and Alessandro Di Nuovo[3]

Computing Department and Advanced Wellbeing Research Centre (AWRC), Sheffield Hallam University, Sheffield S1 1WB, UK
kz4658@exchange.shu.ac.uk
acesjm3@exchange.shu.ac.uk
acesad5@exchange.shu.ac.uk

Abstract. This work presents an access control mechanism framework for MARs in healthcare that addresses some of the security and privacy challenges posed by the deployment of such systems. The proposed system presents a novel authorisation mechanism, to achieve fine-grained zero-trust access control while preserving the confidentiality of patient data. The proposed mechanism incorporates privacy-preserving access protocols to prevent unauthorized access attempts and protect the identity of users accessing the system. Including user-centric access control and an authorisation mechanism according to the stakeholder's roles and the requested information at a given time. Additionally, the work introduces the internal process framework to the proposed system, in ROS level of communication.

Keywords: Access control framework · Privacy by design · Security by design.

1 Introduction

The integration of multi-modal assistive robotic systems (MARs) in healthcare has shown great potential support for improving independent living, providing personalized care, improving patient outcomes, and reducing healthcare costs. However, the deployment of MARs in healthcare and the development of human-robot interactions also raise significant concerns regarding the security and privacy of sensitive patient data. In this context, access control mechanisms play a critical role in ensuring that only authorized users can access and manipulate patient data.

2 Background Literature

The traditional one-time authorisation opens access to intruders and potential unauthorised users, therefore, it is crucial to secure the channel of communication to preserve the user's right to privacy. The acceptable use of technology and

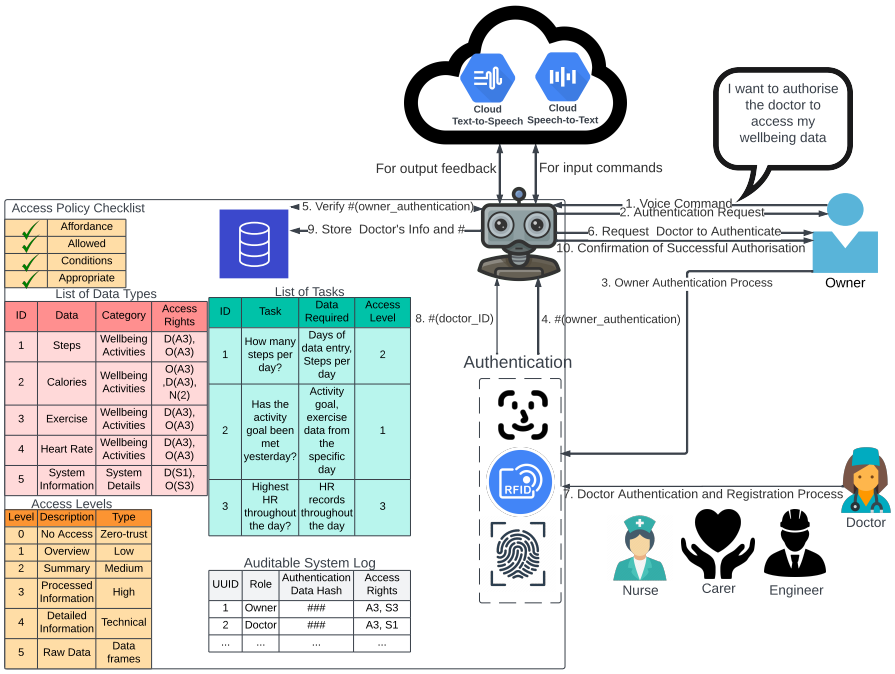
perception of the elderly for monitoring daily living was reviewed by [2], a set of semi-structured interviews conducted on adults, resulting in a positive perception of the need for assistive technology. Furthermore, the recent pandemic has given us a range of feedback on the effective adoption of digital solutions as mentioned in a recent systematic review [3].

Policy development and defining controls can structure the boundaries for maintaining patient's privacy [6], this is also beyond informational privacy as a patient may feel a social-emotional bond with their assistive robot and share information based on their psychological influence [4]. Ethical and privacy questions arise on what data should be accessed by the stakeholders. How much of the data is accessed at a given time? What could impact the data access? What is the retention period of the access? Who authorises the access? And more questions to follow. A recent study discusses access policies, the author claims to contribute first attempts on the architecture and enforcement layer as well as on joint modelling for assistive robots, the author considers challenges, a road map for the policy, objectives layer and activity-centric access control. The paper explores the relationship between task planning, and access control and how this may be adopted in a humanoid assistive robot [1].

A research conducted on state-of-the-art literature on data security and user privacy in interactive social assistive multimodal robotic systems (ARMSys) for healthcare [5]. Analysing the occurrence of data leakage and privacy concerns in a multi-modal setting. The research covers some key security and privacy requirements of ARMSys, suggesting how to improve users' trust by making the decision processing transparent using technologies such as blockchain. The paper continues to discuss authentication challenges, access control and a secure-by-design approach, which could be personalized depending on the type of data and personal preferences. The proposed security framework solution is theorized but not specified such as the use of blockchain, its consensus mechanism and the practicality of the internal process within such system.

3 Proposed Framework

The proposed access control mechanism in Fig. 1 illustrates a novel approach to processing privacy and security by the robot, introducing an access control policy that subsequently verifies the robot's availability of the requested information, whether the authenticated user is authorised to access the data, conditions of the situation to affirm the purpose of the requested information is either based on normal or emergency states and as a sequel whether it is appropriate to share the information requested. Fig. 2 presents the proposed internal process framework within MARs following the robot operating system's (ROS) communication channels. The nodes within the framework represent the access control mechanism using ROS nodes and communication methods.



ID	Data	Category	Access Rights
1	Steps	Wellbeing Activities	D(A3), O(A3)
2	Calories	Wellbeing Activities	O(A3) ,D(A3), N(2)
3	Exercise	Wellbeing Activities	D(A3), O(A3)
4	Heart Rate	Wellbeing Activities	D(A3), O(A3)
5	System Information	System Details	D(S1), O(S3)

ID	Task	Data Required	Access Level
1	How many steps per day?	Days of data entry, Steps per day	2
2	Has the activity goal been met yesterday?	Activity goal, exercise data from the specific day	1
3	Highest HR throughout the day?	HR records throughout the day	3

UUID	Role	Authentication Data Hash	Access Rights
1	Owner	###	A3, S3
2	Doctor	###	A3, S1
...	...	...	...

Level	Description	Type
0	No Access	Zero-trust
1	Overview	Low
2	Summary	Medium
3	Processed Information	High
4	Detailed Information	Technical
5	Raw Data	Data frames

Fig. 1. Proposed Authorisation and Access Control Mechanism Framework

4 Analysis and Discussion

The system logs are designed to enhance accountability and auditability of the decision-making process, considering the inputs and outputs of the system for the user-robot interaction, based on the authorised level of access in addition to the role-based access. The proposed system is for a realistic user-centric healthcare scenario where the assistive robot holds the patient's wellbeing data and the stakeholders can retain only the necessary information by direct communication to the MARs, the findings indicate possible improvement to independent living in a healthcare environment by delegating the burden of responding to wellbeing related questions to the robot, as a result, reducing the human error factor. The system logic from the user input to the action process is handled by the controller node, the sole responsibility of the node is to direct the overall process flow of the system in place. User inputs are processed by the interpreter node, sent to the controller node to confirm the authentication and pass to the access policy node where the checklist process takes place. The final outcome of the process is either granted or rejected and the feedback is given back to the user. The system communicates through the channels in plain text by default, therefore the security features must be enabled to prevent security incidents.

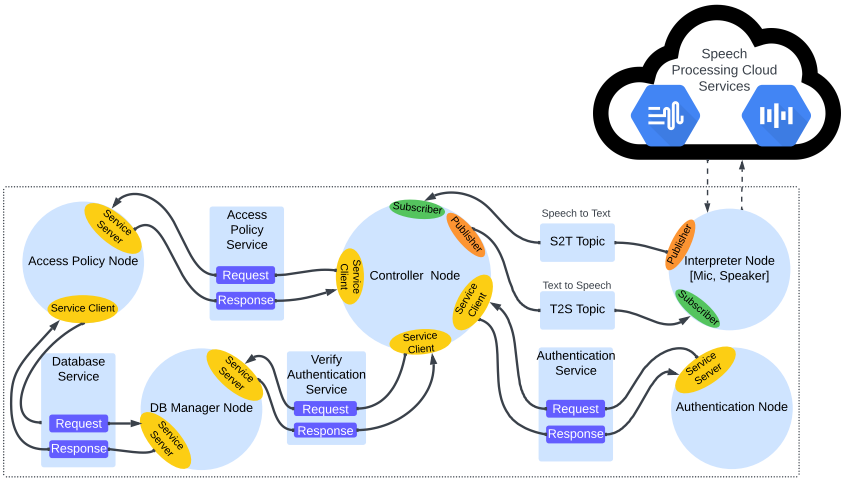


Fig. 2. Proposed Robot Operating System (ROS) Internal Process Framework

5 Conclusion

Our approach can be used as a blueprint for the design and development of secure and privacy-preserving access control mechanisms, an auditable log system and an authorisation process for MARs in healthcare and other sensitive domains.

References

1. Bayreuther, S., Jacob, F., Grotz, M., Kartmann, R., Peller-Konrad, F., Paus, F., Hartenstein, H., Asfour, T.: Bluesky: Combining task planning and activity-centric access control for assistive humanoid robots. In: Proceedings of the 27th ACM on Symposium on Access Control Models and Technologies. pp. 185–194 (2022)
2. Camp, N., Johnston, J., Lewis, M.G., Zecca, M., Di Nuovo, A., Hunter, K., Magistro, D.: Perceptions of in-home monitoring technology for activities of daily living: semistructured interview study with community-dwelling older adults. JMIR aging **5**(2), e33714 (2022)
3. Golinelli, D., Boetto, E., Carullo, G., Nuzzolese, A.G., Landini, M.P., Fantini, M.P., et al.: Adoption of digital technologies in health care during the covid-19 pandemic: systematic review of early scientific literature. Journal of medical Internet research **22**(11), e22280 (2020)
4. Lutz, C., Schöttler, M., Hoffmann, C.P.: The privacy implications of social robots: Scoping review and expert interviews. Mobile Media & Communication **7**(3), 412–434 (2019)
5. Marchang, J., Di Nuovo, A.: Assistive multimodal robotic system (amrsys): security and privacy issues, challenges, and possible solutions. Applied Sciences **12**(4), 2174 (2022)
6. Meingast, M., Roosta, T., Sastry, S.: Security and privacy issues with health care information technology. In: 2006 International Conference of the IEEE Engineering in Medicine and Biology Society. pp. 5453–5458. IEEE (2006)

Personalised Interactive Reinforcement Learning with Multi-Task Pre-training*

Imene Tarakli and Alessandro Di Nuovo

Sheffield Hallam University, Sheffield, United-Kingdom i.tarakli@shu.ac.uk

Abstract. This paper explores the role of multi-task pre-training in enhancing the efficiency of Interactive Reinforcement Learning (RL) for personal robots. By employing multi-task pre-training, the robot is initially trained on a subset of environments and then personalised by users through evaluative feedback in unseen settings of the environment. The preliminary results demonstrate that pre-training significantly improves sample efficiency, facilitating quicker convergence to personalised behaviours. These results demonstrate a promising approach to enhancing adaptation and personalisation in the field of social robotics and enhancing the usability of personal robots across diverse domains.

Keywords: Robotic Personalisation · Interactive Reinforcement Learning · Evaluative feedback.

1 Introduction

Personal robots have emerged as a highly promising technology for carrying out interactive tasks in various settings, including domestic and public environments. A critical factor in ensuring the successful adoption of these robots on a widespread scale is the ability to achieve a high level of personalisation in their behaviour, allowing them to be tailored to the specific needs and preferences of individual users [13]. However, attaining such a degree of personalisation presents significant challenges, which can potentially be addressed through the inclusion of human users in the control loop.

One approach that facilitates user-driven personalisation is Interactive Reinforcement Learning (RL). In this method, users provide direct feedback to the robot, actively influencing its decision-making [10]. One specific approach of interactive RL is through teaching with evaluative feedback; users personalise the robot's behaviour by assessing each performed action[14]. In this approach, humans with non-programming skills can transfer their task knowledge to the robot by providing evaluative feedback, which reduces the robot's exploration time and speed up its learning compared to classical RL[**?**]. This approach has been proven successful in real-world robot control [5] [11] [8]. However, the process requires evaluating a large number of actions before arriving at an effective

* This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 955778.

policy, which becomes particularly challenging during the initial stages when the robot is initialized with a random policy. Consequently, providing accurate feedback becomes difficult for users.

To address this challenge, certain studies [6] propose initialising the robot's policy with an expert demonstration provided by the human user. This approach helps reduce the amount of human feedback needed for convergence. However, it may prove to be very challenging for non-expert human trainers to deliver high-quality demonstrations [7]. In this study, we investigate the role of pre-training in improving the sample efficiency of Interactive RL. Our approach focuses on multi-task pre-training, where the robot is initially taught a general policy in a subset of pre-defined environments [12]. Users can then personalise the robot's behaviour in new environments through evaluative feedback. Figure 1 provides an overview of the proposed framework. By leveraging pre-training, our goal is to enhance the learning efficiency of Interactive RL and enable more effective user-driven personalisation in robotic systems.

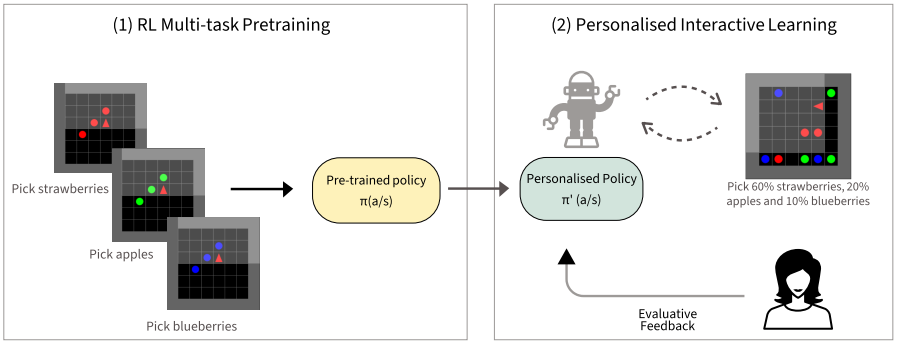


Fig. 1: Illustration of the framework. (1) The robot first learns a basic policy by performing a multi-task RL pretraining on different environments. (2) A user then personalises the robot's behaviour on a new task through evaluative feedback.

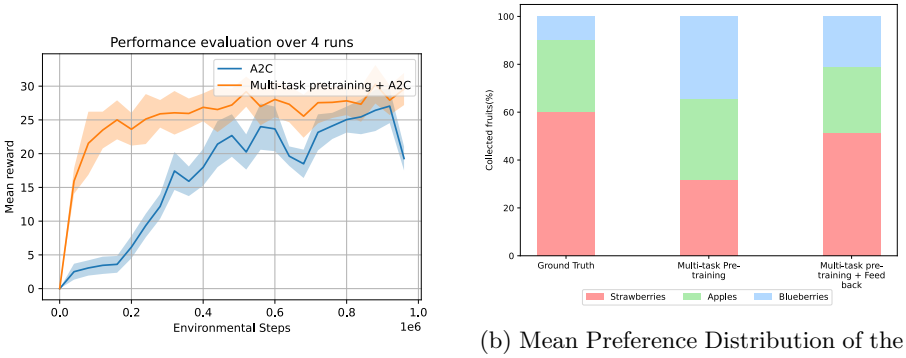
2 Methodology

The idea behind this study is to create an initial decision-making ability for the robot with multi-task RL pre-training, that can later be personalised by users through feedback. This approach enables the robot to handle different tasks that share the same state and action space but have distinct goals defined by separate reward functions. By employing this method, we hypothesise that the agent will learn a better representation of the environment that would allow users to rapidly personalise the robot to a new unseen configuration of the environment.

We consider the Fruit-Picking environment [1] [3]. It consists of a gridworld containing different types of fruits that the robot needs to collect. Similar to previous work [1], the robot learns to pick one type of fruit per task during the pre-training task. In the personalisation phase, users are required to teach the agent a different task that was not previously encountered by expressing varying degrees of preference for each fruit in the environment (for e.g. :60% preference for strawberries, 30% preference for apples, and 10% preference for blueberries). We train the agent using a deep Advantage Actor-Critic Architecture. [2].

Figure 2a presents the results obtained from training an A2C agent on the Fruit-Picking environment, comparing the performance with and without multi-task RL pretraining. Our approach (see orange line) achieves the same performance using 8× fewer environmental steps. This demonstrates that the multi-task pre-training can enhance the sample and feedback efficiency of interactive learning with evaluative feedback.

Additionally, Figure 2b displays the preference distribution achieved after training the robot to collect fruits based on the user preferences. We notice that the initial multi-task pre-training enables the robot to collect all types of fruits equally. After training with the simulated feedback, the robot learns a policy (51.4% strawberries, 27.3% apples, 21.3% blueberries) that proportionately corresponds to the actual distribution of ground truth (60% strawberries, 30% apples, 10% blueberries). This suggests that our approach allows the robot to learn policies that align with the human's intent, enabling users with no programming expertise to personalise the robot behavior with fewer feedback.



(a) Learning curve of A2C agent with and without pre-training on the Fruit Picker task.

(b) Mean Preference Distribution of the collected fruits on 1000 episodes of the Fruit Picker task.

Fig. 2: Evaluation of the framework on a new unseen environment.

3 Discussion

Multi-task RL pre-training has several benefits for personalisation and efficient interactive learning for personal robots. The inclusion of pre-training notably enhanced sample efficiency, supporting the hypothesis that it enables the robot to acquire a comprehensive environmental representation, accelerating learning when incorporating human feedback. Additionally, the pre-training phase facilitated the robot's adaptation to user preferences, allowing users to refine the robot's behavior through evaluative feedback to align it more closely with the desired distribution.

Future work should consider comparing the effectiveness of multi-task RL pre-training with alternative methods, such as pre-training from demonstration [4] and meta RL [9]. Additionally, we will validate the framework on physical robots with human participants which will allow us to evaluate the scalabilty of the method with real human feedback.

References

1. Abdulhai, M., Jaques, N., Levine, S.: Basis for intentions: Efficient inverse reinforcement learning using past experience. arXiv preprint arXiv:2208.04919 (2022)
2. Babaeizadeh, M., Frosio, I., Tyree, S., Clemons, J., Kautz, J.: Reinforcement learning through asynchronous advantage actor-critic on a gpu. arXiv preprint arXiv:1611.06256 (2016)
3. Chevalier-Boisvert, M., Bahdanau, D., Lahlou, S., Willems, L., Saharia, C., Nguyen, T.H., Bengio, Y.: Babyai: A platform to study the sample efficiency of grounded language learning. arXiv preprint arXiv:1810.08272 (2018)
4. de la Cruz, G.V., Du, Y., Taylor, M.E.: Pre-training with non-expert human demonstration for deep reinforcement learning. The Knowledge Engineering Review **34**, e10 (2019)
5. Knox, W.B., Stone, P.: Framing reinforcement learning from human reward: Reward positivity, temporal discounting, episodicity, and performance. Artificial Intelligence **225**, 24–50 (2015)
6. Li, G., He, B., Gomez, R., Nakamura, K.: Interactive reinforcement learning from demonstration and human evaluative feedback. In: 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN). pp. 1156–1162. IEEE (2018)
7. Lin, J., Ma, Z., Gomez, R., Nakamura, K., He, B., Li, G.: A review on interactive reinforcement learning from human social feedback. IEEE Access **8**, 120757–120765 (2020)
8. MacGlashan, J., Ho, M.K., Loftin, R., Peng, B., Wang, G., Roberts, D.L., Taylor, M.E., Littman, M.L.: Interactive learning from policy-dependent human feedback. In: International Conference on Machine Learning. pp. 2285–2294. PMLR (2017)
9. Mandi, Z., Abbeel, P., James, S.: On the effectiveness of fine-tuning versus meta-reinforcement learning. arXiv preprint arXiv:2206.03271 (2022)
10. Najar, A., Chetouani, M.: Reinforcement learning with human advice: a survey. Frontiers in Robotics and AI **8**, 584075 (2021)

11. Najar, A., Sigaud, O., Chetouani, M.: Training a robot with evaluative feedback and unlabeled guidance signals. In: 2016 25th IEEE international symposium on robot and human interactive communication (RO-MAN). pp. 261–266. IEEE (2016)
12. Oh, J., Singh, S., Lee, H., Kohli, P.: Zero-shot task generalization with multi-task deep reinforcement learning. In: International Conference on Machine Learning. pp. 2661–2670. PMLR (2017)
13. Tarakli, I., Angelopoulos, G., Hellou, M., Vindolet, C., Abramovic, B., Limongelli, R., Lacroix, D., Bertolini, A., Rossi, S., Di Nuovo, A., et al.: Social robots personalisation: At the crossroads between engineering and humanities (concatenate). In: Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction. pp. 920–922 (2023)
14. Thomaz, A.L., Breazeal, C., et al.: Reinforcement learning with human teachers: Evidence of feedback and guidance with implications for learning performance. In: Aaai. vol. 6, pp. 1000–1005. Boston, MA (2006)

Towards an Abstract Lightweight Multi-robot ROS Simulator for Rapid Experimentation*

Laurence Roberts-Elliott[1][0000−0001−9877−0553], Gautham P. Das[1][0000−0001−5351−9533], and Alan G. Millard[2][0000−0002−4424−5953]

[1] Lincoln Agri-Robotics, University of Lincoln, United Kingdom
{lrobertselliott,gdas}@lincoln.ac.uk
[2] Department of Computer Science, University of York, United Kingdom,
alan.millard@york.ac.uk

Abstract. Modern robot simulators are commonly highly complex, offering 3D graphics, and simulation of physics, sensors, and actuators. The computational complexity of simulating large multi-robot systems in these simulators can be prohibitively high. To achieve faster-than-realtime simulation of a multi-robot system for rapid experimentation, we present 'move_base_abstract', a ROS package providing a high-level abstraction of robot navigation as a "drop-in" replacement for the standard 'move_base' navigation, and a bespoke integrated minimal simulator. This bespoke simulator is compatible with ROS and strips the simulation of robots down to the representation of robot poses in 2D space, control of robots via navigation goals, and control of simulation time over ROS topic messages. Replication of an existing MRS simulated study using 'move_base_abstract' executed 2.87 times faster than the real-time that was simulated in the study, and analysis of the results of this replication shows room for further optimisations.

Keywords: Abstract Simulation · Multi-Robot Systems Simulation · Robot Navigation.

1 Background

Most popular robot simulators, such as Gazebo and Webots, can boast 3D graphics, and simulation of physics, sensors, and actuators. This detailed simulation is necessary for many applications, but it has performance costs that can make faster-than-real-time simulation difficult. Some applications, however, may not require such complexity of simulation. For many MRSs (Multi-Robot Systems), representation of time, robot poses, and robot navigation, is sufficient for initial broad exploration or comparison of algorithms and methods [2,1]. Behaviours that robots may engage in once they reach allocated tasks may be abstracted as simply having a robot wait for the expected task duration. Task behaviours may otherwise be implemented by the simulator's user, with more niche robot tasks less likely to have a readily available implementation. While this high level

* This work was supported by the UKRI's E3 fund via Lincoln Agri-Robotics.

of abstraction may not present as reliable an analogue of a real-world MRS as a more complex simulation might, its simplicity enables experimentation at a pace unattainable within more realistic simulators, especially when the complexity of simulations often scales poorly with increasing numbers of robots. Abstract simulators can enable researchers of MRSs to investigate a broader range of possible solutions for their applications, the best of which can be simulated more accurately in a realistic simulator to increase confidence in these methods before committing to time-expensive implementation in a real-world MRS.

We present 'move_base_abstract' (MBA), a Free and Open-Source Software (FOSS) Robot Operating System (ROS) compatible "drop-in" replacement for the standard move_base navigation ROS package [3], with a bespoke built-in minimalist simulator aimed at maximising performance, but also compatible with the 'stage_ros' and 'Gazebo' simulators, for users requiring more detailed simulation. It can be accessed at GitHub.com/laurencejbelliott/move_base_abstract. MBA's simulator aims to provide faster-than-realtime simulation at speeds higher than headless Gazebo, or other 2D simulators such as stage_ros and Flatland, by abstracting to a level above simulation of sensors, physics such as collisions and forces, and complex robot navigation. Use cases for MBA are exemplified in a ROS port of the multi-robot soil properties mapping simulation described in Section 2.2, and in a "simulation mode" for the Innovate UK Robot Highways project's systems. The soil mapping MRS simulation is not concerned with simulating static obstacle avoidance or detailed physics, as these are not required to represent an open field. Nor does it require simulation of sensor data except for soil compaction measurements. Robot Highways' "simulation mode" uses MBA in conjunction with topological navigation and its bespoke coordinator, to perform static obstacle avoidance by constraining navigation along predefined paths, and for traffic negotiation to avoid collision or deadlocking with other robots, respectively.

2 Methodology

2.1 move_base_abstract (MBA)

MBA replaces the 'move_base' navigation stack with a high-level abstraction of robot navigation and provides a bespoke integrated minimalist simulator. Upon receiving a navigation goal, this abstracted navigation system essentially calculates the time that would be required for the robot to reach the goal moving in a direct line, multiplying Euclidean distance by a given constant speed. It then waits for the calculated time to elapse in ROS time, and finally updates the robot's pose to the pose specified in the navigation goal. The abstracted navigation system is interfaced as a 'ROS Action' compatible with the commonly used 'move_base' ROS Action, allowing for sending, monitoring, and cancelling goals in a standardised manner. The bespoke simulator is designed to leverage standard ROS utilities, such as 'map_server', 'RViz', and control of ROS time. MBA can also publish to topics used to update robot poses in 'stage_ros' and 'Gazebo', if the user needs a more detailed simulator.

2.2 Replication of Multi-Robot Soil Compaction Mapping Trials

In a previous work, 'Agent-Based Simulation of Multi-Robot Soil Compaction Mapping' [5], we developed a simulation of a MRS for mapping soil compaction in an open field using the Mesa agent-based simulation Python library [4]. Simulated trials assessing the performance of different configurations of a multi-robot soil compaction mapping system were previously conducted in this Mesa-based simulation, and we replicate these in a ROS compatible simulation that uses MBA to vastly reduce the execution time of robot navigation. Currently the 'stage_ros' simulator is used in conjunction with the abstracted navigation of MBA, but future work will see this use MBA's built-in simulator. In the ROS version of the simulation, the simulated trials were run at 3 speeds: 1x, 10x, and 15x, to measure an over-estimation of time elapsed during execution of complex algorithms in the MRS's coordinator. Simulation speed was reduced to 1x real-time during processes such as task allocation and task creation to reduce this over-estimation. To measure the similarity of results between different runs of these simulated trials, the cosine similarity was calculated between the mean performance metrics from one run of the simulated trials, and the same from another run, as shown in Table 1.

3 Results and Analysis

Table 1. Mean of cosine similarities calculated across metrics recorded in pairs of simulated multi-robot soil mapping studies.

Name of 1st Set of Trials	Name of 2nd Set of Trials	Mean Cosine Similarity
TAROS 22 Mesa trials	TAROS 23 stage_ros trials - 10x sim. speed	0.969960079
TAROS 23 stage_ros trials - 1x sim. speed	TAROS 23 stage_ros trials - 10x sim. speed	0.992865229
TAROS 23 stage_ros trials - 1x sim. speed	TAROS 23 stage_ros trials - 15x sim. speed	0.869587149

Cosine similarity ranges from -1 to 1, with values closer to 1 indicating greater similarity. As seen in Table 1, a relatively high cosine similarity ≈ 0.97 was measured when comparing the performance metrics from the Mesa trials with those from the ROS trials run at 10x speed. This shows that the Mesa simulation trials can be replicated in the ROS-based simulation with very similar results. Some small dissimilarity may be explained by assumptions of the Mesa simulation that are not made in the ROS simulation. These include Mesa operating in discrete integer steps of time (seconds in our case), and continuous processes being run once every time step. This means that complex computation that can take several seconds to execute always takes 1 second in Mesa's time. The mean cosine similarity appears to decrease as the simulation speed is increased. The

results remain highly similar to 1x speed at 10x speed, but are relatively dissimilar at 15x speed. At 10x speed, the ROS simulation trials executed 2.87x faster than realtime overall. At 15x speed, this increased to 14.18x times faster than realtime, but the results in some metrics were significantly different to those measured at 1x speed.

4 Conclusions and Future Work

At 10x speed, the results of the stage_ros simulation trials are very similar to those seen when running the simulation at wall time. It may be necessary to further optimise the simulator and MRS coordinator's more complex Python code, or port some of the more complex nodes to C++ (starting with the coordinator). This could help to attain reasonably accurate results when running the simulation at speeds beyond 10x. The simulated experiments were run in a Parallels virtual machine on a 2020 MacBook Pro with an 8-core 3.2GHz Apple M1 CPU, and 16GB of RAM. It is expected that running simulated trials on "bare metal" on a faster CPU could enable faster simulated time speeds, and this may be investigated in future work. Future work may also support ROS 2, and simulate sensors and collisions in 2D, with the option to enable/disable these features as necessary to trade performance for fidelity. This simulation, and the bespoke simulator, provide a closer analogue to a real-world multi-robot system than Mesa, both by making fewer assumptions, and through their use of the ROS middleware. This should require minimal changes to the software to enable it to coordinate real robots with ROS.

References

1. Choudhury, S., Gupta, J.K., Kochenderfer, M.J., Sadigh, D., Bohg, J.: Dynamic multi-robot task allocation under uncertainty and temporal constraints. Autonomous Robots **46**(1), 231–247 (Jan 2022). https://doi.org/10.1007/s10514-021-10022-9, https://doi.org/10.1007/s10514-021-10022-9
2. Das, G., Cielniak, G., From, P., Hanheide, M.: Discrete Event Simulations for Scalability Analysis of Robotic In-Field Logistics in Agriculture – A Case Study. Brisbane (May 2018), https://eprints.lincoln.ac.uk/id/eprint/32170/
3. Marder-Eppstein, E., Berger, E., Foote, T., Gerkey, B., Konolige, K.: The Office Marathon: Robust navigation in an indoor office environment. In: 2010 IEEE International Conference on Robotics and Automation. pp. 300–307. IEEE, Piscataway, New Jersey, USA (May 2010). https://doi.org/10.1109/ROBOT.2010.5509725, iSSN: 1050-4729
4. Masad, D., Kazil, J.: Mesa: An Agent-Based Modeling Framework. pp. 51–58 (Jan 2015). https://doi.org/10.25080/Majora-7b98e3ed-009
5. Roberts-Elliott, L., Millard, A.G., Das, G.P.: Agent-Based Simulation of Multi-Robot Soil Compaction Mapping. In: To appear in Lecture Notes in Computer Science. p. 15. Springer, Oxford, UK (Sep 2022)

Automating Robotic Gas Distribution Mapping in Unknown and GPS-denied Environments

Mal Fazliu[1] and Cunjia Liu[1] [0000-0002-3121-7208]

[1] Department of Aeronautical and Automotive Engineering, Loughborough University, Leicestershire, UK, LE11 3TU
m.fazliu-19@student.lboro.ac.uk
c.liu5@lboro.ac.uk

Keywords: Mobile robotics, Gas distribution mapping, informative path planning.

Introduction

Rapid and precise acquisition of situational information during a hazardous gas release is crucial to emergency responses. Mobile robotic gas distribution mapping (GDM) is an emerging technology that provides first responders with real-time spatio-temporal data through gas concentration map media. However, the current operation of these robotic platforms remains heavily dependent on teleoperation. By automating these tools, one can eliminate the requirement of having trained remote specialists on site and facilitate optimized GDM.

Previous studies on autonomous GDM often suffer from incompleteness or impracticality in terms of attaining a fully automated system capable of addressing real-world hazardous release scenarios. For instance, the sensor simulated (non-robot) informative path planning study in [1] and the novel K-means clustering solution proposed in [2] perform GDM under a pre-determined occupancy grid map (OGM). A mobile robot however, when faced with a new hazardous scene, will not have access to such prior knowledge; the robot instead will need to build its own OGM using the Simultaneous localization and mapping (SLAM) functionality. Sampling strategies for real-world applications therefore would also need to actively infer both gas map and SLAM information to achieve a practical, optimized solution. Studies like [3] which attempted such work fell short of practical success in complex, obstacle-tight environments because of their use of the Kernel DM+V GDM function whose computational efforts leave little room for higher quality path planning, path following, SLAM and goal selection algorithms to be installed.

In [4], Rhodes et al. addresses the feasibility issue of 3D structurally-aware GDM by introducing a Gaussian Belief Propagation (GaBP) factor graph solver to the Gaussian Markov Random Field (GMRF) gas representation first proposed by Monroy et al. in [5], which was initially recognized because of its accuracy and superior plume-obstacle modelling properties. Their novel hybrid message scheduling system in [4] has enabled computationally feasible, real-time 2D and 3D GDM in unknown and obstacle rich environments to emerge, but the sensing robot is still teleoperated.

This study aims to develop an autonomous GaBP-GDM framework with the path planning and following function for mobile robotic applications seeking to operate in unknown and GPS denied environments, supporting the deployment of vehicles in both real life and high-fidelity simulation.

Autonomous GDM Framework Development

The general system architecture for an autonomous GDM platform has been included in Fig.1 to familiarize the reader with the subject.

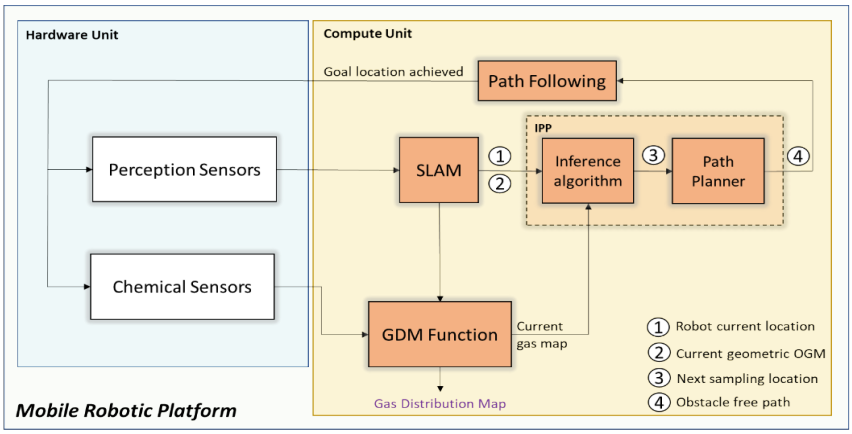


Fig. 1. General system architecture for an autonomous GDM application. The key functional components in the Compute Unit include SLAM to support robotic navigation and OGM building, the GDM function from [4] and the newly developed robot planning function.

To fully test and verify different functions, the integrated simulation environment is developed. The framework is achieved in the Robotics Operating System (ROS) and a Clearpath Husky is employed during the study as the mobile sensing robot. To

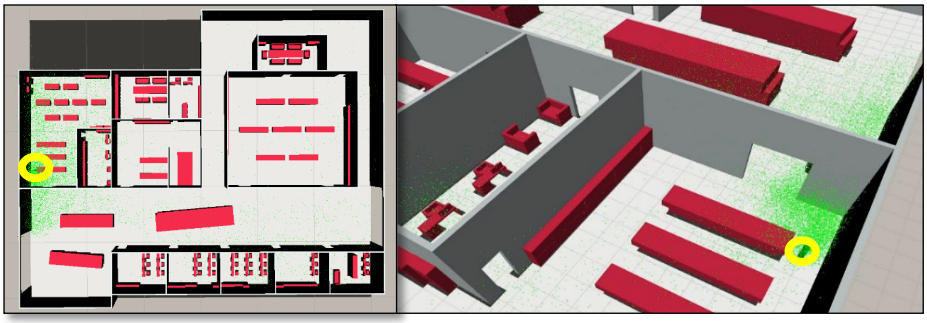


Fig. 2. Gas dispersion visualized in RViz via GADEN playback node.

test the autonomous platform, a custom 60 x 50m factory environment gas leak scenario was created using GADEN [6], a ROS high fidelity gas dispersal simulator

(Fig.2). Tight aisles were intentionally included. Note that an equivalent Gazebo setting was created to act as the real world and mimic any physics interactions like robot-obstacle collisions.

Established and commercially available ROS packages which provide functionality within path planning, low level path following control, as well as SLAM were explored. The final set of packages chosen are highlighted in Table 1. Google cartographer SLAM, known for its computationally efficient mapping capabilities in large environments, was key in providing accurate localization, especially when compared to the default EKF localization system provided by Clearpath which would tend to cause drift greater than 30 cm after several minutes of operation. Cartographer, on the other hand, ensured drift would not exceed approximately 10 cm and importantly would reduce error at times because of the loop closure characteristics offered by the function. Unlike the proposed platform in [3], our framework does not rely on GPS for localization. Only LiDAR and other SLAM compatible sensors are needed, as well as a suitable chemical sensor, all of which can be configured by a user via ROS nodes. An informative path planning (IPP) solution is used to select the next sampling location which can provide more information gain. Move Base was another essential component to the framework as it provided robust path planning and control techniques, including Dijkstra for global path planning, Dynamic Window Approach (DWA) for local, and a high-level velocity-based control, for the robot to navigate to the next gas sampling location.

Table 1 Key ROS packages implemented.

	SLAM	**Path Planning**	**Path Following**
Package	Cartographer	Move Base (Dijkstra, DWA)	Move Base (Velocity High level)

A custom inference algorithm which communicates with the rest of the framework was also created in the form of several ROS nodes. The inference network was designed with mechanisms in place to ensure that the robot does not seek an unreachable goal. In the IPP solution, several inference techniques that utilized live gas concentration mean, uncertainty and processed frontier data from live SLAM were explored.

Results

Each inference algorithm's performance was evaluated against a lawn-mower ground truth sweep by measuring the RMSE of the modelled gas distribution over time. To conduct GDM, PID sensors with update rates of 1 Hz were used to capture the chemical measurements. The GaBP mapping algorithm was able to incorporate the new measurements and update its media within 0.5 second. Some visuals, including a final gas map from the autonomous GaBP simulation are provided in Fig. 3 to show the capability of the developed simulation environment and the robotic GDM capability.

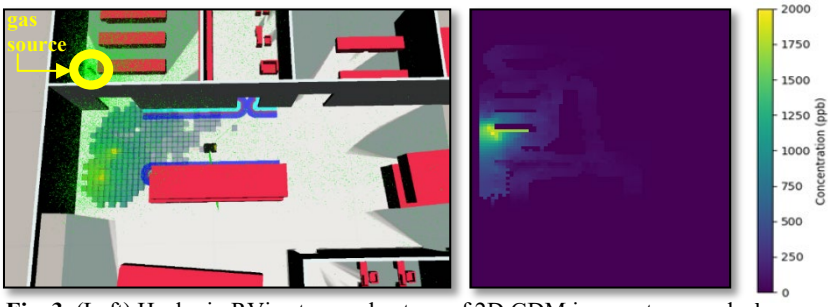


Fig. 3. (Left) Husky in RViz at an early stage of 2D GDM in a custom gas leak scenario. (Right) The final gas map achieved at the end of the robotic GDM study.

Conclusions

With the recent introduction of the GaBP factor graph solver, robotic GDM platforms are now subject to real-time inference updates that can be made available through an informative path planner to generate strategic destinations. This study addresses this opportunity by introducing a custom ROS framework that provides ready automation for mobile robotic GaBP-GDM applications operating in unknown, cluttered GPS-denied environments. In the context of future IPP algorithm development for GDM, the ROS framework introduced offers to greatly reduce testing time.

References

[1] C. Rhodes, C. Liu, and W. H. Chen, "Informative path planning for gas distribution mapping in cluttered environments," *IEEE International Conference on Intelligent Robots and Systems*, pp. 6726–6732, Oct. 2020.

[2] C. Ercolani *et al*., "Clustering and Informative Path Planning for 3D Gas Distribution Mapping: Algorithms and Performance Evaluation," *IEEE Robot Autom Lett*, vol. 7, no. 2, pp. 5310–5317, Apr. 2022.

[3] Y. A. Prabowo, B. R. Trilaksono, E. M. I. Hidayat, and B. Yuliarto, "Utilizing a Rapidly Exploring Random Tree for Hazardous Gas Exploration in a Large Unknown Area," *IEEE Access*, vol. 10, pp. 15336–15347, 2022.

[4] C. Rhodes, C. Liu, and W.-H. Chen, "Structurally Aware 3D Gas Distribution Mapping Using Belief Propagation: A Real-Time Algorithm for Robotic Deployment," *IEEE Transactions on Automation Science and Engineering (Early Access),* pp. 1–15, 2023.

[5] J. G. Monroy, J. L. Blanco, and J. Gonzalez-Jimenez, "Time-variant gas distribution mapping with obstacle information," *Auton Robots*, vol. 40, no. 1, pp. 1–16, Jan. 2016.

[6] J. Monroy, V. Hernandez-Bennetts, H. Fan, A. Lilienthal, and J. Gonzalez-Jimenez, "GADEN: A 3D Gas Dispersion Simulator for Mobile Robot Olfaction in Realistic Environments," *Sensors 2017*, vol. 17, no. 7, pp. 1479–1495, Jun. 2017.

Towards Heterogeneous Modular Robotic Systems for Industrial Applications

Joshua W. G. Gilmour[0009−0001−8841−6355], Edward Bray[0009−0004−7110−4787], and Roderich Groß[0000−0003−1826−1375]

Department of Automatic Control and Systems Engineering, The University of Sheffield, Sheffield, UK {jwggilmour1,enbray1,r.gross}@sheffield.ac.uk

Abstract. This paper presents HeteroMod, a heterogeneous modular robotic system designed for industrial use. HeteroMod comprises core modules and add-ons that expand the workspace and capabilities of the system. It allows the axes of two core modules to be combined, by which the torque output can be increased. To demonstrate the potential of HeteroMod, a prototype system is built and tested.

Keywords: Modular Robotics · Reconfigurable Systems · Industry

1 Introduction

Modular robots offer a flexible approach to building robotic systems, where multiple modules are combined into different physical configurations [1, 3]. Such systems could choose and autonomously produce configurations that suits their tasks. This could be especially relevant in industry, where there is a need to rapidly adapt to changing task demands and manipulate large payloads. The modular systems in [2, 5] could be applied in these industrial settings, however they are restricted to manipulator configurations, and their maximum payload is limited as modules cannot be attached in parallel to combine their torques.

This paper presents HeteroMod, a novel modular robotic system for industrial applications. HeteroMod uses a combination of active and passive elements to produce a wide range of configurations. Modules can also be combined in parallel to effectively double the lift capability of individual modules.

2 Design

The HeteroMod concept (Fig. 1a) is built around *core* modules (c) with additional capabilities provided by *add-on* modules, including links (l), and end-effectors (e). The structures they can form are described using grammar rules:

$$S \to cX; S \to lS; X \to cX; X \to lX; X \to e,$$

where S is the start symbol. This formulation assumes that core and add-on modules can occur in an arbitrary order, and that at least one active module

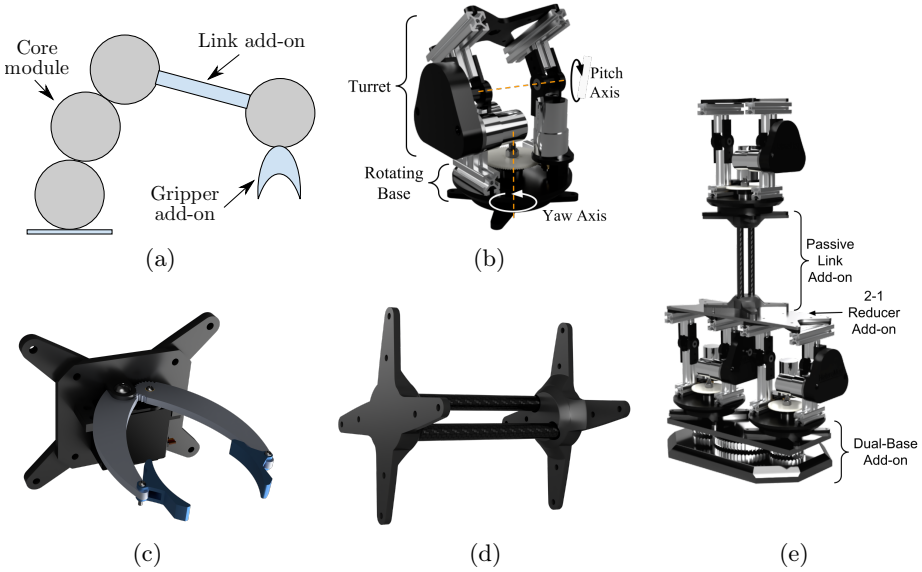
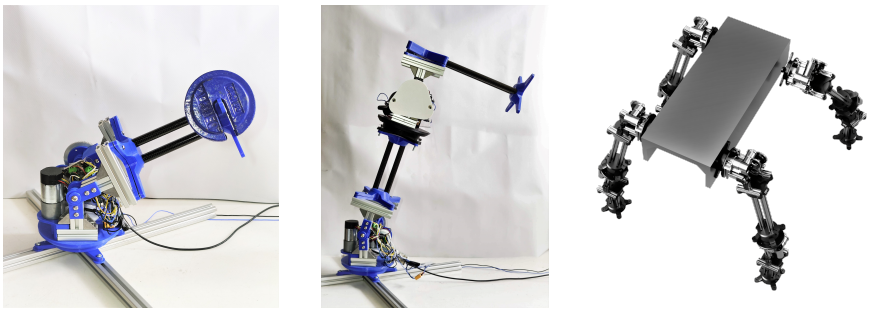


Fig. 1. HeteroMod system. (a) The concept, and (b) the design of the core module. Example add-on modules include (c) a gripper, (d) a straight link, and (e) the combination of a dual-base and 2-to-1 reducer.

is required to transmit power to the end-effector. Further add-ons combine core modules in parallel: these are a dual-base (d) and 2-to-1 reducer (t). With slight abuse of notation, we write $S \rightarrow d \begin{bmatrix} c \\ c \end{bmatrix} tX$, where terminal symbol $\begin{bmatrix} c \\ c \end{bmatrix}$ denotes the parallel presence of two core modules.

The core module (Fig. 1b) utilises a central base plate, which the electronics and top *turret* section attach onto. The base plate is rotated by a single motor to provide *yaw* movement. The turret features a motorised hinge mechanism providing *pitch* movement, and houses dual motors and control boards.

Add-on modules that consume power are termed *active*, otherwise they are *passive*. Currently, a powered gripper module providing basic grasping capability has been designed (Fig. 1c). Other potential active add-ons could provide further actuation capabilities such as dispensing from syringes, or include sensing technologies such as cameras. Passive add-ons allow the creation of a wider range of topologies. Two passive link add-ons have been designed: a straight carbon-fiber link (Fig. 1d) and a right-angled link. They reduce the number of active modules required to form large structures, hence decreasing the weight and cost of such configurations. Minimising the weight in this manner reduces the torque required to actuate the configuration, further widening the range of potential applications. Further passive modules could include a vehicle base to utilise the yaw actuation for wheeled locomotion.



(a) (b) (c)

Fig. 2. System evaluation: (a) physical 2 DoF manipulator used in the lift capacity trials, (b) physical 4 DoF manipulator configuration, and (c) concept quadruped walker.

For larger manipulator configurations, core modules are combined through a dual-base and 2-to-1 reducer (Fig. 1e). This provides a strong base and high torque capabilities for heavier payloads. To the best of our knowledge, it is the first adaptor for a modular robotic system that allows multiple axes of multiple modules to be combined. The dual-base adaptor utilises a combination of gears attached to a rotating plate to combine the yaw axes of the modules, while the 2-to-1 reducer consists of a single plate to combine the aligned pitch axes.

The majority of the custom parts are 3D printed from PLA+ plastic to allow rapid prototyping and minimise development costs. Geared DC motors with encoders for rotation sensing provide movement of the joint axes through a further 5:1 gear reduction. On the pitch axis, a pair of motors provide a theoretical maximum stall torque output of 44 Nm after additional reduction.

Physical connections between modules are made through standardised adaptor plates. Currently, wing-nuts and bolts allow for rapid reconfiguration without tools. Future work will develop a powered quick-connect module to enable automatic reconfiguration.

Wireless communication for controlling the modules is implemented using the reliable Message Queuing Telemetry Transport protocol, which allows untethered control, suitable for locomotion configurations. A basic closed loop PID controller is used to move the module axes to the desired positions based on the motor encoder feedback. Power is provided to the core modules via an external power supply, providing 12 V DC. Future work will explore power sharing between modules, as well as battery-powered operation.

3 Testing and Potential Uses

To validate system performance, lift capacity tests were carried out on a prototype core module (Fig. 2a). A load was moved from a horizontal to vertical orientation while lifting masses of 0.5 kg, 1.0 kg, 1.5 kg, 2.0 kg and 2.5 kg. The maximum torque output on the pitch axis of an active module was measured as

5.9 Nm before gears slipped: this compares favourably with the 3.07 Nm produced by the similar HyMod system [4]. The motors are capable of producing significantly more torque than demonstrated in these trials, which could be utilised by exchanging the 3D printed pitch gear mounting plate for a more rigid component.

To show the generalisability of HeteroMod, several configurations were created. These are a prototype manipulator with a theoretical payload capacity of 6.7 kg and a reachable workspace diameter of 1.15 m (Fig. 2b), as well as a conceptual quadruped walker (Fig. 2c) with a passive 'body' add-on. Furthermore, an articulated platform to transfer collections of parts between distinct locations was created. Potential applications include manipulation of a range of payloads on assembly-lines, or repeated visual inspection of hard-to-reach components with a camera add-on. These could benefit from platforms that can rapidly adapt to bespoke torque and reachability requirements, and that support the integration of custom add-ons.

4 Conclusion

This paper has presented HeteroMod, a heterogeneous modular robotic system based on homogeneous active modules. The system is scalable to a number of configurations and can be produced at low cost: the current price of materials for each core module is around £200. Although certain components could benefit from reinforcement, the underlying principle and design has been shown effective through prototype modules and initial testing. When compared to [2, 5], HeteroMod offers greater flexibility for end-effector options and user-designed add-on modules. Additionally, a parallel attachment method for improved torque at a given joint allows for greater scalability than current systems. Future work will refine the mechanical design, as well as investigate automatic coupling and power sharing between modules to realise a cost-effective solution for industrial applications.

References

1. Jorgensen, M., Ostergaard, E., Lund, H.: Modular ATRON: modules for a self-reconfigurable robot. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). vol. 2, pp. 2068–2073 (2004)
2. Liu, S.B., Althoff, M.: Optimizing performance in automation through modular robots. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 4044–4050 (2020)
3. Murata, S., Kakomura, K., Kurokawa, H.: Toward a scalable modular robotic system. IEEE Robotics & Automation Magazine **14**(4), 56–63 (2007)
4. Parrott, C., Dodd, T.J., Groß, R.: HyMod: A 3-DOF hybrid mobile and self-reconfigurable modular robot and its extensions. In: Distributed Autonomous Robotic Systems, Springer Proceedings in Advanced Robotics, vol. 6, pp. 401–414. Springer (2018)
5. Yun, A., Moon, D., Ha, J., Kang, S., Lee, W.: ModMan: An advanced reconfigurable manipulator system with genderless connector and automatic kinematic modeling algorithm. IEEE Robotics and Automation Letters **5**(3), 4225–4232 (2020)

Angled Cantilever Construction by Force-Aware Robotic Swarms

James Kellett, Edward Bray[0009−0004−7110−4787], and Roderich Groß[0000−0003−1826−1375]

Department of Automatic Control and Systems Engineering, The University of Sheffield, Sheffield, UK {jwkellett,enbray1,r.gross}@sheffield.ac.uk

Abstract. Swarms of robots could be used to self-assemble into bridges across gaps in their terrain, with potential applications including in disaster relief. Prior work has shown how such bridges can be constructed in a force-aware manner to ensure they do not collapse, but only considers structures with a flat upper surface. We build on this work by considering how structures that follow a specified path, for example between platforms at two different heights, can be safely constructed. This increases the versatility of these structures, bringing the research closer to the requirements of real-world application.

Keywords: Swarm robotics · Cantilevers · Force sensitive robot

1 Introduction

By taking inspiration from ants [1], researchers have developed several distributed control schemes to self-assemble bridges from robots [2, 3, 5]. This could be beneficial to allow robots to explore new areas in unpredictable environments, for example when large groups of small robots are deployed to search underneath the rubble of a collapsed building. Past researchers have shown how force-aware methods can be used during such self-assembly to ensure these structures will not collapse, allowing for safe construction in the real-world [3, 4]. These works consider the construction of cantilevers with a flat upper surface, so can only form bridges across gaps where both sides are the same height. In this paper, we consider how the agents can be influenced to construct a cantilever at a given angle to the horizontal, or through more complex paths. This adds versatility for applications where the target location to build towards is unknown, or where obstacles are present between the initial and target locations.

2 Problem Formulation

This work considers a similar environment to previous studies by Bray and Groß [3]. Simulated square agents occupy a 2D grid of side length 0.1 m. They construct a cantilever from a vertical fixed support on the left of the environment, as shown in Fig. 1. Agents can separately sense the moment M and axial force

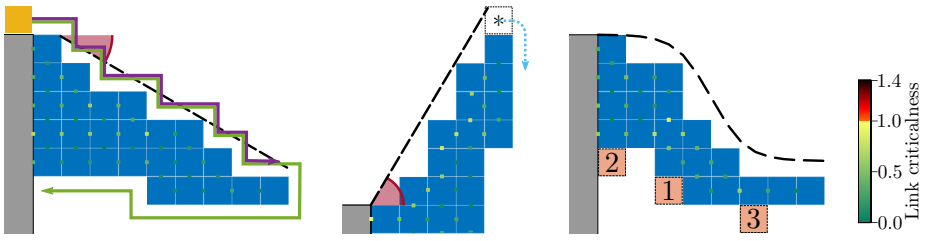


Fig. 1. The stages of the algorithm. Placed agents are shown in blue, active agents in yellow, and the fixed support in grey. The desired paths by black dashed lines: in (a) and (b) these are defined by the red angles, while (c) shows a non-linear path. (a) The active agent first moves around the structure communicating with placed agents, following either the purple or green path in the message-passing or local variants respectively. (b) If the active agent believes the structure is stable, it extends the cantilever along the path by placing in the first possible location clockwise from above the rightmost column (cyan line emanating from the asterisked location). (c) If the active agent believes the structure is unstable, it reinforces the structure: locations 1 and 2 are valid, whereas location 3 is invalid.

F in the *links* they make to adjacent agents. These are compared to *allowable limits* to calculate a *criticalness* for each link by dividing each measurement by its allowable limit and taking the maximum resulting value; we assume links are sufficiently strong in compression that negative F can be ignored. When either allowable limit is exceeded, the link is close to failure and described as *critical*. Structures with no critical links are referred to as *stable*, else they are *unstable*.

Active agents initialise above the fixed support, and move around the perimeter of the structure, communicating with their neighbours to receive information about the current state of the force distribution within the structure. Once they have sampled all the columns, they add themselves to the structure in a column of their choice. They are now *placed*, and can no longer move. Another active agent initialises when the previous one has placed itself in the structure. This process repeats until a specified number of agents have been added.

The goal of the algorithm is to produce a stable cantilever that follows a given target path for the longest distance while using the fewest number of agents: this path could either be a straight line at an angle to the horizontal (Figs. 1a & 1b), or a more complex function (Fig. 1c). The forces within links are calculated using a truss-based simulator, which excludes the weight of the active agent [3].

3 Algorithm Design

The distributed self-assembly algorithm begins with the active agent gathering data about the distribution of internal forces within the structure. Prior work investigated two different methods of communicating the values of M and F

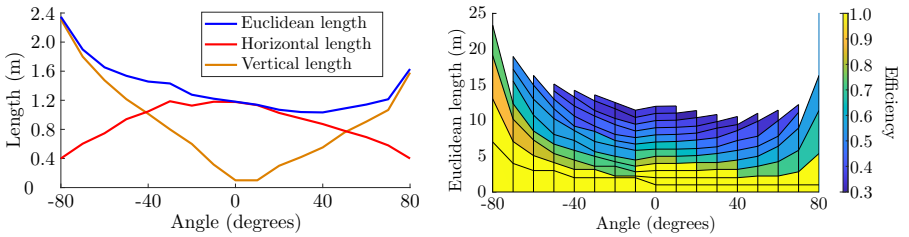


Fig. 2. Simulation results. (a) The average Euclidean length reached at each angle, and the corresponding horizontal and vertical lengths. (b) The efficiency of structures produced for each angle and Euclidean length.

measured in links of placed agents, called the *message-passing* and *local* variants [3]: we also investigate the differences between these approaches in this new scenario. In the message-passing variant, placed agents coordinate to communicate the maximum values of M and F measured in each column to the agent at its top. The active agent therefore must only travel along the top of the surface and communicate with these top agents to obtain this information (Fig. 1a, purple line). The local variant removes this communication between placed agents, and instead requires the active agent to travel all the way around the structure (Fig. 1a, green line) to obtain the measurements made by all the placed agents on the perimeter, as these will theoretically be the largest in each column.

When the active agent has received the necessary force information, it must choose where to place itself within the structure. If it believes the structure is stable, it will place itself to extend the cantilever along its predefined path by either placing in the rightmost column, or by extending into a new column. Beginning with the location above the rightmost column (marked with an asterisk in Fig. 1b), agents check each location in a clockwise order to find the first one that does not extend beyond the desired path boundary, but is both empty and will produce a contiguous structure. If the active agent believes the structure is unstable, it calculates a probability distribution as described in [3] that represents the probability of placing in each column to provide reinforcement, where columns with critical links are more likely to be chosen. The active agent draws from this distribution without replacement and attempts to place at the bottom of the chosen column, but can only do so if there is a placed agent to either side: locations 1 and 2 in Fig. 1c are therefore valid, whereas location 3 is not. If an invalid column is chosen, another column is chosen without replacement.

4 Results

We performed systematic trials on cantilevers with straight paths at a given angle θ to the horizontal, as shown in Figs. 1a & 1b for negative and positive θ respectively. Trials were performed for linear paths with $-80° \leq \theta \leq 80°$ in $10°$

intervals, with the allowable M and F set to 13.9 Nm and 579 N respectively. A total of 30 trials were performed for each θ, each terminating when 40 agents were placed.

The cantilevers are compared by their maximum *Euclidean length*, which refers to the straight line distance from the upper right corner of the fixed support to the tip along the surface path. Fig. 2a shows the average maximum length reached for each θ, and reveals a global minimum at $\theta = 30°$, whereas one might expect this to occur when building horizontally ($\theta = 0°$). A combination of factors contributed to this result. In the absence of gravity, structures close to $\theta = \pm 45°$ consisting of a given number of agents are of the lowest Euclidean length due to the discrete grid environment. The inclusion of gravity means structures require reinforcement, which is more efficient for negative θ as the structure can form a buttress against the fixed support with fewer agents.

An efficient cantilever will travel the furthest distance using the least possible agents. Fig. 2b shows the *efficiency*, a metric calculated as the Euclidean length of a structure divided by number of agents it comprises. The plot shows that the larger angles are capable of maintaining the highest efficiency for a given Euclidean length. The small angles are initially efficient but exhibit a large decline as θ increases, explained by the rapid increase in forces as the distance from the fixed support increases. This analysis is irrespective of agent size.

5 Conclusion

This paper has shown how swarms of force-aware robots can self-assemble structures beyond those presented in literature. It proposes a distributed algorithm by which robots build structures at a prescribed angle. Such angled structures were analysed in detail; the same method was also shown to build structures along a non-linear path. The structures produced did not collapse during construction and could be chosen to follow a wide range of paths, increasing the potential scenarios such robots could be deployed in.

References

1. Anderson, C., Theraulaz, G.: Self-assemblages in insect societies. Insectes Sociaux **49**, 99–110 (May 2002)
2. Andrés Arroyo, M., Cannon, S., Daymude, J.J., Randall, D., Richa, A.W.: A stochastic approach to shortcut bridging in programmable matter. Natural Computing **17**(4), 723–741 (2018)
3. Bray, E., Groß, R.: Distributed self-assembly of cantilevers by force-aware robots. In: 2021 International Symposium on Multi-Robot and Multi-Agent Systems (MRS). pp. 110–118. IEEE (2021)
4. Inou, N., Fukushima, S., Shimotai, N., Ujihashi, S.: Study of Group Robots Adaptively Forming a Mechanical Structure : Effect of Mechanical Properties of Cellular Robots on Structure Formation. JSME Int. Journal Series C **43**(1), 127–133 (2000)
5. Malley, M., Haghighat, B., Houe, L., Nagpal, R.: Eciton robotica: Design and Algorithms for an Adaptive Self-Assembling Soft Robot Collective. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 4565–4571 (2020)

Prototyping Mechanical Design of a Robotic Tail for Human Balance Support

Rafi Hossain, Eisa Anwar, Mohammad Allama Irfan,
Theeran Krishna Sithanen, Diar Jalal, Faizan Hussain,
Antonino Cafiero-Regueira, Muhammad Mustafa Shahzad, Ashish Ahluwalia,
Sajeeva Abeywardena, and Ildar Farkhatdinov

Queen Mary University of London, UK

Abstract. This short paper presents the results of a final year project carried out by a group of university students, wherein the design and prototyping of a wearable robotic tail for enhancing human balance was investigated. The main goal of the research was to explore the two-degrees of freedom kinematic configuration of the robotic tail and to determine the power and actuation requirements involved. Through iterative experimentation, a prototype of the tail was successfully constructed and tested. The prototype incorporates a two-link mechanism, which is affixed to a user's back for support. Cable-driven actuation and electric motors are utilized at the ends of each mechanical link of the tail to achieve the desired actuation.

Keywords: wearable robotics, balance control, human-robot interaction

1 Introduction

Musculoskeletal injuries resulting from improper lifting techniques in various work environments present significant challenges, both in terms of physical well-being and economic costs [1]. Wearable robots can be used to support human posture and balance during material handling tasks improving safety and efficiency of the work [2]. Limited solutions for wearable robotic tails have been proposed earlier [3–5] however, only one of the prototypes [4] was tested for balancing support. Our previous simulation studies have demonstrated that it is possible to use a wearable robotic tail attached to a user's back to support balancing tasks when handling heavy loads [6–8]. In this work we explored the design and prototyping of a wearable robotic tail that can support human posture during heavy objects handling with a potential application with supernumerary robotic limbs, as shown in Fig. 1a.

2 Mechanical Design of the Robotic Tail

The prototype consists of a flange designed to be fixed on a user's back using a commercially available back-frame support. It comprises two cable-driven revolute joint units with ball-bearing units and pulley stacks for the cable-driven

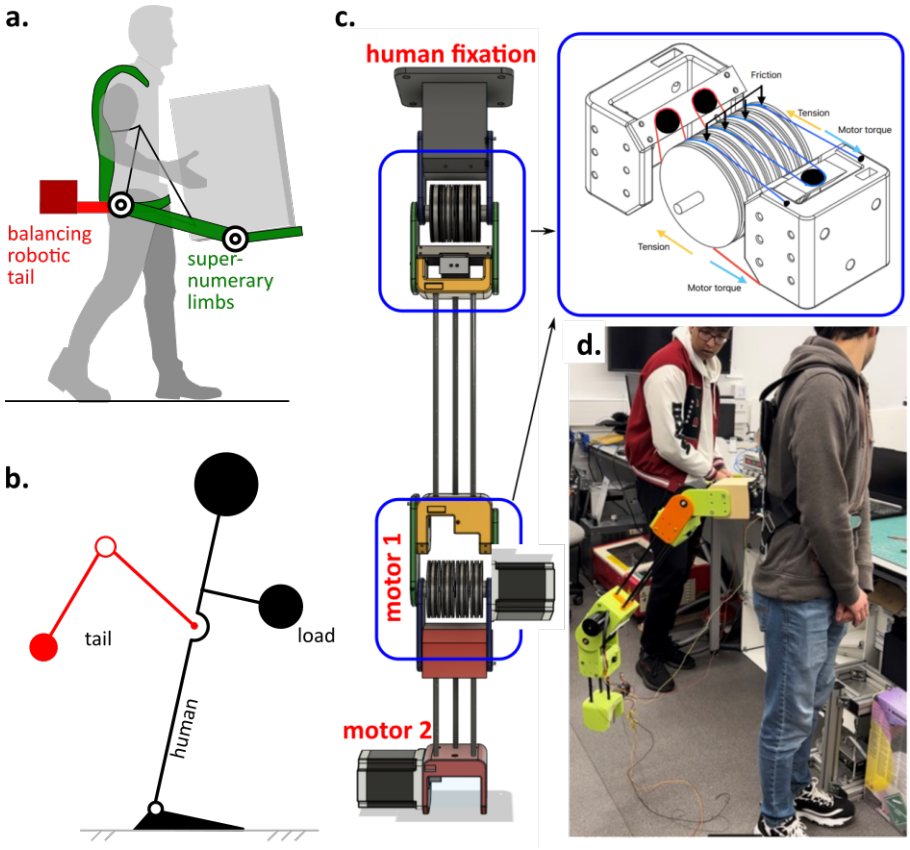


Fig. 1: **a.** Conceptual drawing of a robotic tail for human balancing support during material handling with supernumerary robotic limbs. **b.** Kinematic scheme of 2-DoF robotic tail and a simplified human balancing model. **c.** CAD of the robotic tail prototype. Top-right: cable mechanism used for rotation transmission in the joints. **d.** Manufactured robotic tail worn by a user during the tests.

mechanisms. Both revolute joint units are designed using similar 3D-printed parts, ensuring modularity and simplified maintenance. The tail is mechanically attached using three carbon fiber rods, with motors positioned at the second (middle joint) and the tail's end to achieve a distant center of mass. The lengths of the tail's links are 70 cm and 40 cm, respectively, and the total weight of the robot does not exceed 7 kg.

To maintain balance and minimize the risk of backward falls, it was crucial to ensure that the center of mass of the tail does not extend beyond 23 cm from the user's back in its passive configuration (vertically facing down). This constraint maintains a balanced weight distribution and reduces the potential moment around the user. The calculated torque for motor 1 is 0.625 N/m, and for motor 2, it is 4 N/m, both of which fall within the maximum torque limits of the selected brushless direct current actuators. The use of a high transmission ratio in the cable drive system (20-30 times torque reduction depending on the number of pulleys used) allows relatively low torque requirements. However, the power requirements for the motors are relatively high to ensure the tail is fast and reactive in counter-balancing tasks. The calculated power capacity for motor 1 is 573 W, and for motor 2, it is 262 W.

Fig. 1d depicts a user wearing the robotic tail during initial testing, attached to an off-the-shelf wearable back support. In testing with a healthy adult male user, the tail successfully supported its weight when positioned in a fixed configuration. It was also demonstrated that the tail effectively influenced human posture, requiring the user to adjust their center of mass position to compensate for the tail's movement. This postural support will be crucial when testing the tail in load-carrying tasks.

3 Conclusion

This paper presented a mechanical design and prototype manufacturing of a wearable robotic tail that can be used to support human balance and posture in material handling tasks. Several design aspects were discussed and potential design and application challenges identified. The robotic tail was validated in a test with a healthy adult male participant and it was demonstrated that the reaction torques created by the tail movement are sufficient to influence a users balance control.

One of the key challenges faced in this research is the power requirements of the system. To ensure the tail can accelerate to the necessary balancing configuration quickly, while accounting for human body dynamics and posture control, a substantial amount of torque is needed. Currently, the system relies on external power sources, which were not integrated into the tail's design, presenting a significant limitation.

For successful future applications of this robotic technology, it is essential to consider a more compact design and a simplified mechanical configuration. Addressing these aspects can lead to an improved and more efficient wearable robotic tail system.

References

1. S. M. Spence, G. M. Jensen, and K. F. Shepard, "Comparison of methods of teaching children proper lifting techniques," *Physical Therapy*, vol. 64, no. 7, pp. 1055–1061, 1984.
2. I. Farkhatdinov, J. Ebert, G. Van Oort, M. Vlutters, E. Van Asseldonk, and E. Burdet, "Assisting human balance in standing with a robotic exoskeleton," *IEEE Robotics and automation letters*, vol. 4, no. 2, pp. 414–421, 2019.
3. H. Xie, K. Mitsuhashi, and T. Torii, "Augmenting human with a tail," in *Proceedings of the 10th Augmented Human International Conference 2019*, pp. 1–7, 2019.
4. A. Maekawa, K. Kawamura, and M. Inami, "Dynamic assistance for human balancing with inertia of a wearable robotic appendage," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4077–4082, IEEE, 2020.
5. J. Nabeshima, M. Y. Saraiji, and K. Minamizawa, "Prosthetic tail: Artificial anthropomorphic tail for extending innate body functions," in *Proceedings of the 10th Augmented Human International Conference 2019*, pp. 1–4, 2019.
6. S. Abeywardena, E. Anwar, S. Miller, and I. Farkhatdinov, "Human balance augmentation via a supernumerary robotic tail," in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 2878–2881, IEEE, 2022.
7. S. Abeywardena, E. Anwar, S. Miller, and I. Farkhatdinov, "Dynamic assessment of supernumerary tails for balance augmentation,"
8. S. Abeywardena and I. Farkhatdinov, "Towards enhanced stability of human stance with a supernumerary robotic tail," *IEEE Robotics and automation letters*, 2023.