



https://helda.helsinki.fi

Comparison of Semi-autonomous Mobile Robot Control Strategies in Presence of Large Delay Fluctuation

Seppänen, Alvari

2022-09

Seppänen , A , Vepsäläinen , J , Ojala , R & Tammi , K 2022 , ' Comparison of Semi-autonomous Mobile Robot Control Strategies in Presence of Large Delay Fluctuation ' , Journal of Intelligent and Robotic Systems , vol. 106 , no. 1 , 28 . https://doi.org/10.1007/s10846-022-01711-3

http://hdl.handle.net/10138/350665 https://doi.org/10.1007/s10846-022-01711-3

cc_by publishedVersion

Downloaded from Helda, University of Helsinki institutional repository.

This is an electronic reprint of the original article.

This reprint may differ from the original in pagination and typographic detail.

Please cite the original version.

REGULAR PAPER



Comparison of Semi-autonomous Mobile Robot Control Strategies in Presence of Large Delay Fluctuation

Alvari Seppänen^{1,2} · Jari Vepsäläinen¹ · Risto Ojala¹ · Kari Tammi^{1,2}

Received: 15 February 2022 / Accepted: 21 July 2022 / Published online: 17 September 2022 © The Author(s) 2022, corrected publication 2022

Abstract

We propose semi-autonomous control strategies to assist in the teleoperation of mobile robots under unstable communication conditions. A short-term autonomous control system is the assistance in the semi-autonomous control strategies, when the teleoperation is compromised. The short-term autonomous control comprises of lateral and longitudinal functions. The lateral control is based on an artificial potential field method where obstacles are repulsive, and a route is attractive. LiDARbased artificial potential field methods are well studied. We present a novel artificial potential field method based on color and depth images. Benefit of a camera system compared to a LiDAR is that a camera detects color, is cheaper, and does not have moving parts. Moreover, utilization of active sensors is not desired in the particle accelerator environment. A set of experiments with a robot prototype are carried out to validate this system. The experiments are carried out in an environment which mimics the accelerator tunnel environment. The difficulty of the teleoperation is altered with obstacles. Fully manual and autonomous control are compared with the proposed semi-autonomous control strategies. The results show that the teleoperation is improved with autonomous, delay-dependent, and control-dependent assist compared to the fully manual control. Based on the operation time, control-dependent assist performed the best, reducing the time by 12% on the tunnel section with most obstacles. The presented system can be easily applied to common industrial robots operating *e.g.* in warehouses or factories due to hardware simplicity and light computational demand.

Keywords Mobile robots · Teleoperators · Machine vision · Remote monitoring · Automation · Autonomous vehicles

1 Introduction

Teleoperated mobile robots are used in monitoring, inspection, and maintenance tasks, in hazardous or otherwise nonhuman friendly places. The operator is not required on-site, which enables bilateral control over the internet. Remote control introduces additional control and feedback delays to

 Alvari Seppänen alvari.seppanen@aalto.fi
 Jari Vepsäläinen jari.vepsalainen@aalto.fi
 Risto Ojala risto.j.ojala@aalto.fi

Kari Tammi kari.tammi@aalto.fi

¹ Department of Mechanical Engineering, Aalto University, Otakaari 4, 02150 Espoo, Uusimaa, Finland

² Helsinki Institute of Physics, University of Helsinki, 00014 Helsinki, Uusimaa, Finland the system, which can fluctuate due to the nature of internet protocols. The internet connection instability can be caused by weather conditions, network congestion, solar activity, geography, and buildings. Fluctuating control delay makes the teleoperation challenging, which can lead to accidents and slow mission operation.

Teleoperation with fluctuating control delay has been studied before. Delay passivation dampens the control depending on the delay, which has been shown to smooth the control and assist the teleoperation with low delays [1-3]. However, in the case of large delays the control is stopped until the delay is low enough to pass a delay threshold value. The threshold value has ranged from 300 ms up to 2000 ms in previous studies. [1, 4].

Thus, the teleoperation mission takes longer due to the delay and leaves the system helpless in situations where it is critical to keep moving. We present novel autonomously assisted control strategies for high control delay scenarios. These strategies are namely delay-dependent assist (DDA) and control-dependent assist (CDA). DDA activates

autonomous control when the delay threshold is reached and CDA assists the teleoperator when the control is considered inadequate, similar to lane keeping assistant of passenger vehicles. Authors in [5–9] have successfully aided a teleoperator of a robot manipulator using semi-autonomous control. Our aim is to successfully implement a semi-autonomous teleoperation system for a maintenance robot operating at the particle accelerator tunnel of the European Organization for Nuclear Research (CERN). The validation experiments are carried out in a mock-up environment of the tunnel. In these experiments, different teleoperation control strategies (DDA, CDA, delayed manual, manual, and autonomous) are compared under fluctuating control delay. In addition to the contribution of the semi-autonomous control strategies, we propose a novel RGB-D image-based artificial potential field (RGB-DPF) method, which is utilized for obstacle avoidance and route following of the autonomous control.

The purpose of this study is to address the issues caused by a large delay fluctuation in the communication signal, and to complement research gaps in previous studies presented in Section 2. The research gap in the teleoperated systems is the implementation of a semi-autonomous control strategies that improve teleoperation in presence of long and high delay periods (5 s duration of over 300 ms delay). We also introduce and validate the RGB-DPF method, which builds on the work presented in [10]. Our contributions are summarized on the list below.

- We propose a novel semi-autonomous control strategy. The strategy is presented in a form of two variants to address the issues of a large delay fluctuation in the communication signal (5 s periods of over 300 ms delay) in mobile robot applications. The strategy variants are based on alternating between autonomous and manual control depending on the control and communication signals.
- Demonstration that during long cutouts (up to 5 s) the robot can be successfully navigated with a short-term autonomous control system, and the teleoperator can gain control back easily when the cutout ends.
- A new adaptation of the artificial potential field method for an RGB-D sensor coupled with a simple control algorithm for semi-autonomous navigation in the CERN particle accelerator tunnel or similar environment (*e.g.* warehouse, factory facility).
- The overall novelty of this paper is the large-scale comparative evaluation of aforementioned methods on a real robot prototype.

Proposed RGB-DPF method includes an attractive field (AF) and a repulsive field (RF). Their details are described in Sections 3.2 and 3.3. The control strategies are described in Section 4.1. The results are presented in Sections 4.3, 4.4,

and 4.5. Discussion is presented in Section 5. The paper is concluded in Section 6 including future research proposals.

2 Related Work

2.1 Semi-autonomous Systems in Internet-based Teleoperation

Mobile robot teleoperation is well established field. The studies in delay management aim to improve real-time internet-based teleoperation. Delay fluctuations, video feedback delay, and limited bandwidth are the main problems related to internet-based teleoperation [1]. Semi-autonomous control systems have been implemented to aid the teleoperator under communication issues or environmental challenges.

You et al. [5] proposed semi-autonomous control system where the operator controls the velocity of the body of a hexapod robot master, and the slave computes the corresponding angular velocities of the leg joints automatically. The operator receives haptic feedback, which is proportional to the error between the actual velocity and the desired velocity of the body. This system aimed to make the teleoperation more intuitive and accurate. The experiments in a semi-physical simulation validated the proposed system. However, the effects of delay in the control signal were not studied.

Authors in [7, 8, 11] studied the control of a teleoperated semi-autonomous system with time-varying delays and input uncertainties. Their systems were driven by the local and remote robot velocities, and position and velocity tracking errors. They improved the performance of a teleoperated system in a simulation setting. The delay fluctuations were periodic and had high frequency (0.5 Hz), and the effects of long and high delay periods were not studied.

2.2 Artificial Potential Field Method

The artificial potential field (PF) method is a path planning method, where obstacles are repulsive, and the goal is attractive. Virtual repulsive and attractive fields are formed, which sum forms a total field. The path is planned according to descent direction of the total field so that the robot avoids obstacles and reaches the goal. The repulsive field is typically formed directly from range sensor data, where obstacles have potential if their range subceed a threshold or from a map. The method was formulated first by Khatib et al. [12], and it has been later improved and implemented e.g. in: [10, 13–19].

However, the PF method has its problems, one of which is the tendency to get stuck on local stable-points. Yao et al. [15] developed black-hole potential field method for avoiding obstacles and reaching a goal in presence of multiple goals. They use black-hole analogue to avoid local stablepoints. Their method also improved the traditional PF method with artificial intelligence, which purpose is to adapt quickly to scenarios containing new types of obstacles. The purpose of the black-hole is to bend the gradient in order to avoid the local stable-points. Conducted experiments with static and dynamic obstacles and goals showed that the robot learns how to automatically jump out of local stable-points.

Orozco-Rosas et al. [16] used an approach that utilizes evolution rules to solve the local stable-point problem. They implemented a hybrid algorithm based on membrane pseudo-bacterial potential field (MemPBPF). MemPBPF algorithm combines membrane computing, pseudo-bacterial genetic algorithm, and the PF method to generate a smooth and safe path for a mobile robot.

The PF method has also been utilized in road vehicle path planning. Rasekhipour et al. [20] developed a PF-based method that observes road regulations and avoids obstacles with appropriate vehicle dynamics. Simulations in complex scenarios validated that the system was able to plan an optimal path in terms of vehicle dynamics.

Jang-Ho et al. [10] used the PF method as a short-term planner similarly as we. In short-term planner implementations, the PF is derived from sensor data only, and no maps are used. They implemented the obstacle-dependent Gaussian potential field (ODG-PF), which defines the obstacles with Gaussian function. They prevented the local stablepoint issue by using the global minimum of the field as the desired direction. Their simulations and experiments proved stable performance of the ODG-PF with both static and dynamic obstacles. The experiments were conducted with a mobile robot equipped with a 2D-LiDAR.

3 Methods

3.1 System Specifications

The robot used in this experiment is a 3-wheeled differential drive model (Fig. 1). It has approximately 0.9 by 0.9 m footprint, with a rigid steel frame. We have equipped the robot with two typical 350 W electric scooter hub brushless direct current motors. The on-board computer is Jetson TX2 from Nvidia, which has a 2.0 GHz 6-core CPU. The software is written with Python3 and C++ languages. We use containerized Ubuntu 20.04 and ROS Noetic, and OpenCV and NumPy libraries. For perception we use the stereo camera of Intel RealSense D455 depth vision system, which is an RGB-D camera. The camera system is mounted pointing forward and approximately 0.4 m above the ground plane.



Fig. 1 The prototype robot and the testing environment used in this study

3.2 Route Detection and Sttractive Field

We detect a route line, which is on the ground of the particle accelerator tunnel (Fig. 1), and the attractive field (AF) is formed from the detected line. The route line detection comprises of following main steps:

- 1. acquire an RGB color image from the camera system,
- 2. crop into Region Of Interest (ROI),
- 3. transform RGB to HSV color model,
- 4. apply HSV mask and acquire the edges,
- 5. apply Hough line detector, and
- 6. form the AF.

We are detecting the line on the ground, so the ROI is naturally the lower part of the image, given the orientation of the camera system. Maximal horizontal Field Of View (FOV) is used, so the route line is detectable even when the trajectory deviates drastically from the route line (Fig. 2).

The thresholds for HSV mask are found manually, and the objective is to filter out everything except the route line. Binary image is formed where pixels have value 1 if they passed the HSV mask, otherwise their value is 0. Then, Hough line detection algorithm (cv2.Hough-LinesP()) is applied, which gives us the end-point coordinates of the detected lines [21]. We take the two most probabilistic ones and compute the arithmetic mean of the crossing points of these lines and a specified horizontal line. Finally, we form the AF according to this mean with following equation

$$A(p_u) = \beta |p_{goal} - p_u| \tag{1}$$

where β is a scaling parameter defining the magnitude of the field. p_u denotes the horizontal pixel coordinate and p_{goal} denotes the arithmetic mean of the crossing points.



Fig. 2 The ROI for the route detection. On the right, the trajectory deviates drastically from the route line

3.3 Obstacle Detection and Repulsive Field

The obstacles are detected from the depth images. The main steps of the repulsive field (RF) formation are the following:

- 1. acquire a depth image from the camera system,
- 2. extract the ground plane and a threshold, and
- 3. form the RF.

All depth measurements that are above the ground plane and closer than a threshold th = 3m from the camera are obstacles. Hazard of a given obstacle is defined by the magnitude of the RF. The RF is utilized by lateral controller to avoid obstacles. If obstacles are closer than 1.2 m, close range obstacle avoidance (CROA) procedure is executed.

3.3.1 Formation of the Virtual Ground Plane

The ground is visible for the camera system, and we want to differentiate between the ground and obstacles. Therefore, a ground plane extraction procedure is carried out. We use the following assumptions for the camera system and the ground:

- 1. the camera system is an ideal calibrated pinhole camera,
- 2. world and camera coordinate systems are aligned,
- 3. the ground is assumed to be a flat plane, and
- 4. the orientation and the position of the camera system relative to the ground plane stay constant.

The ground plane array is compared to the depth images. The formation of the ground plane array is done once, as we assume that the orientation and position of the camera system relative to the ground plane stay constant. This will reduce the computational cost. The ground plane array has the same size as the depth image, and it includes a projection of a virtual ground plane.

Random sample consensus (RANSAC) plane fit is implemented for defining the virtual ground plane, so it is approximately aligned with the real ground. The plane fit method is applied to the point cloud, which gives us constants a, b, c, and d of Eq. 2 for the virtual ground plane. The equation of the plane is

$$aX + bY + cZ + d - g_{th} = 0 \tag{2}$$

where g_{th} denotes a threshold, which sets the virtual ground plane closer to the camera accounting for depth measurement error, parameter errors, and camera movement. Then we use intrinsic projection matrix (3) for projecting X-coordinate values of plane (2) onto the virtual sensor

$$\lambda \begin{bmatrix} p_u \\ p_v \\ 1 \end{bmatrix} = \begin{bmatrix} f_l & 0 & u_0 \\ 0 & f_l & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Y \\ Z \\ X \end{bmatrix}$$
(3)

where p_u and p_v denote the virtual sensor coordinates in pixels i.e. the indices of the ground plane array, f_l denotes the focal length, and u_0 and v_0 denote the principal point of the virtual sensor. Obstacles have potential if they are above the virtual ground plane and subceed the 3 m threshold. Figure 3 illustrates the virtual ground plane and the threshold.

3.3.2 Repulsive Field

Figure 4 illustrates the nodes and processes of the RF computation. The RF is a 1D-array, which values represent the horizontal potential because the robot is steered in a yaw direction. The full repulsive field is formed from the depth



Fig. 3 The virtual ground plane, X=3 m threshold, and the camera coordinate system. The segment of the obstacle that has potential is marked with red



Fig. 4 The nodes and processes of the repulsive field computation. The numbers indicate the resolution of nodes

image and the ground plane array. This is described by Algorithm 1. The depth image updates with a rate of 30 Hz, and the ground plane array stays constant. It is computed during calibration to reduce computational cost.

Algorithm 1 Ground plane extraction and threshold. 1: $p_h \leftarrow 48\overline{0}$; ▷ Image height 2: $p_w \leftarrow 848;$ ▷ Image width 3: *dIma*: ▷ Depth image 4: gndPln; ▷ Ground plane 5: $fullRF \leftarrow zeros(p_h, p_w);$ ▷ Full repulsive field 6: $th \leftarrow 3000;$ \triangleright The 3 m threshold 7: for $p_v \leftarrow 0: p_h$ do for $p_u \leftarrow 0 : p_w$ do 8: if $dImg[p_v, p_u] < th$ then α. if $dImg[p_v, p_u] < gndPln[p_v, p_u]$ then 10: $fullRF[p_v, p_u] \leftarrow th - dImg[p_v, p_u];$ 11: end if 12:13: else $fullRF[p_v, p_u] \leftarrow 0;$ 14: end if 15: end for 16: 17: **end for**

The full repulsive field is compressed into a 1D-array. That is, the set of *m* largest values of each column of the full repulsive field are computed (m = 30 was noted to work the best during testing). For column *n*, this is expressed as

$$B_{m+1} := \{ b \in K_m : b \ge k \forall k \in K_m \}$$

$$\tag{4}$$

where $K_{m+1} := K_i \setminus B_{m+1}$. We compute the arithmetic mean of this set because the maximum is not stable due to noise in the depth measurement. For column *n*, it is expressed as

$$r_n = \sum_{b \in B_m} \frac{b}{|B_m|}.$$
(5)

The raw repulsive field (RRF) is formed from elements r_n

$$R_{raw}(p_u) = \{r_1, ..., r_{p_w}\}$$
(6)

where the length of the array p_w is the width of the depth image. The global minimum of $R_{raw}(p_u)$ represents the region that has the most space, thus defining the desired direction. However, steering in this direction might lead to collisions because the robot has width. Therefore, obstacles are enlarged in such way that desired direction shifts so that the robot will not collide with them. Figure 5 illustrates this process. A safety threshold s_t is added to the width of the robot w_r for safety reasons, which results in $w = 2s_t + w_r$. Adaptive dilation $a_d(p_u)$ represents this enlargement on the virtual sensor, which depends on the focal length f_l , and the depth $D_o(p_u)$. Here, $D_o(p_u) = th - R_{raw}(p_u)$ to obtain the dilation



Fig. 5 Top-down view of the obstacle enlarging process

$$a_d(p_u) = \begin{cases} \frac{f_l \cdot w}{2(th - R_{raw}(p_u))}, & \text{if } R_{raw}(p_u) (7)$$

The RF $R(p_u)$ is compiled using Algorithm 2. Finally, the potential field (PF) is computed as an element-wise sum of the AF and the RF

$$P(p_u) = A(p_u) + R(p_u).$$
(8)

Alg	orithm 2 RRF adaptive dilation (obstacle enlargement).	
1: .	RRF;	
2: .	$RF \leftarrow zeros(p_w);$	
3: .	$DS \leftarrow f_l * w/2;$	
4: 1	$th \leftarrow 3000;$	
5: İ	for $p_u \leftarrow 0 : p_w$ do	
6:	$startIndex \leftarrow p_u - DS/(th - RRF[p_u]);$	
7:	$stopIndex \leftarrow p_u + DS/(th - RRF[p_u]);$	
8:	$\mathbf{if} \ startIndex < 0 \ \mathbf{then}$	
9:	$startIndex \leftarrow 0;$	
10:	end if	
11:	$\mathbf{if} \ stopIndex > p_w \ \mathbf{then}$	
12:	$stopIndex \leftarrow p_w;$	
13:	end if	
14:	for $i \leftarrow startIndex : stopIndex \ do$	
15:	$\mathbf{if} \ RRF[p_u] > RF[i] \ \mathbf{then}$	
16:	$RF[i] \leftarrow RRF[p_u];$	
17:	end if	
18:	end for	
19: end for		

3.4 Autonomous Control

Figure 6 presents the flow chart of the autonomous control system. Close range obstacle avoidance (CROA) utilizes binary parameters Turning Direction (TD) and Safety Zone (SZ), which are defined from the RF. The RF is divided into two halves, which define the value of TD. In particular, if the average potential of left half is smaller than the one of right half, TD is set to 0, which manifests as counterclockwise turning. SZ indicates if any obstacle is closer than 1.2 m *i.e.* in the SZ. When SZ is 0, the linear speed is at maximum, and the robot controlled laterally according to the PF global minimum. **Lateral control input** is calculated with pure pursuit algorithm [22], resulting in



Fig. 6 The flow chart of the autonomous control system

$$\dot{\theta}_{input} = -v_{input} \frac{2y_{Lg}}{x_{Lg}^2 + y_{Lg}^2}.$$
(9)

The unit of $\dot{\theta}_{input}$ is rad/s. v_{input} denotes the linear speed, which unit is m/s. y_{Lg} and x_{Lg} denote the coordinates of short-term goal point on the ground plane in meters. The global minimum of PF is projected on the ground plane using projection matrix (3) and (2), resulting in aforementioned parameters y_{Lg} and x_{Lg} . Figure 7 illustrates the position of the short-term goal point and the lateral control input.

3.5 Localization

The localization of the robot enables the evaluation of the performance of the proposed control strategies. Therefore, it is crucial to have an accurate and reliable localization system. We use wheel odometry and inertial measurement unit (IMU) data for deriving the location of the robot relative to the starting point. The location is described with 2D Cartesian coordinates, and the heading angle of the robot. Fusion of IMU and wheel odometry data is done with *efk_localization_node* from *robot_localization* ROS package [23], which utilizes extended kalman filter producing more reliable location information compared to using only IMU or wheel odometry. We select the parameters of the filter in such a manner that heading is mainly derived from IMU data and translation from wheel odometry.

3.6 Communication and Delay Generator

Tunnel environments are challenging from a communication point of view. Cutouts and disturbances of the communication signal are caused by the shape of a tunnel, which is narrow and long. These problems are caused by an effect called multipath propagation and can be eased by implementing multiple nodes in the network. Multipath propagation is apparent especially in tunnel environments [24], and



Fig. 7 The short-term goal point when obstacles are absent (left), and present (right)

is caused by constructive and destructive interference of the communication signal. The original signal reflects off the tunnel walls resulting in multiple signals that have phase differences. Interference manifests as loss of signal or delay fluctuations in the communication signal [25]. [26]

The robots in CERN facilities are connected to 4G/3G/ WiFi network [27]. We use WiFi for communication in our test environment, and the distances between the robot, teleoperator computer, and the WiFi module are short (1-15 m). Therefore, the quality of service is excellent consistently. The communication delay is 20 ms on average with standard deviation of 10 ms. We measure the delay during the experiments.

To study the effects of delay fluctuation, we add artificial delay profiles to the control signal. They are added only to the control signal, as the feedback is direct eyesight. Figure 8 presents the applied delay profiles, which are time-dependent. They are periodic and they have different Gaussian noise levels (no autocorrelation).

3.7 Manual Control

We use a typical joystick controller. For the manual control, linear speed range is [0, 0.5] m/s, and angular speed range is [-0.8, 0.8] rad/s. Linear speed is controlled with vertical and angular speed is controlled with horizontal tilt angle of the joystick. The main author is the teleoperator in the experiments because of the experience with the system and ability to control the robot consistently. Consequently, results where the robot is controlled manually are subjective. The feedback in the teleoperation set-up is direct eyesight, thereby the feedback delay is neglected.

3.8 System Nodes

Figures 9 and 10 illustrate the software communication diagrams of the system used in the experiments. Teleoperation site is connected to the robot via WiFi. ROS message topics are denoted by /< topic name >. The messages on the topic /cmd_vel include



Fig. 8 The delay profiles which are applied to the manual control signal in the delay generator node

linear and angular speed of the robot, /delay includes control signal delay, and /odom includes the position and the heading angle of the robot on a plane relative to the starting position.

4 Experiments

Section 4.1 describes the control strategies and the detailed set-up for the testing environment. The testing procedure used in this study is presented in Section 4.2. The results regarding the proposed RGB-DPF method, trajectories, and performance are presented in Sections 4.3, 4.4, and 4.5, respectively.

4.1 Set-up

We carry out a set of experiments for comparing the control strategies, and validating the RGB-DPF method. The experiments are carried out in a testing environment, which mimics the particle accelerator tunnel with several obstacle arrangements. Different control strategies are used to discover which one is the smoothest. Figure 11 presents the test tracks and their obstacle arrangements. The tracks are 15 m long and approximately 2 m wide. The obstacles are cardboard boxes with dimensions of 0.68, 0.38, 0.38 m.

The control strategies are introduced in Table 1. The strategies are Manual, Delayed manual, Autonomous, Delay-Dependent Assist (DDA), and Control-Dependent Assist (CDA). DDA and CDA tests are done with teleoperating, except when delay or control triggers activate the autopilot. The delay profiles are applied to the manual control commands on strategies Delayed manual, DDA, and CDA.



Fig. 9 The software communication diagram of the prototype system



Fig. 10 The communication diagram of the autopilot

DDA and CDA are semi-autonomous strategies. DDA takes over the controls when the delay of the control signal exceeds 300 ms. The reason for this value is that the control strategy usually changes to "control-and-wait" when the delay is larger than 300 ms according to [4]. Therefore, the goal for this system is to aid the teleoperator in this scenario. CDA compares the control commands of the teleoperator and the autopilot. If the difference exceeds a threshold, autopilot is enabled.

4.2 Testing Procedure

In total, we conducted 60 test runs. Trajectory, time, video feed of the on-board camera, and the PF were recorded with rate of 30 Hz. The robot was accelerated to a maximum of 0.5 m/s speed before the start line. Timer starts at the start line and stops at the finish line. Thus, a theoretical minimum completion time for a mission is 30 s, as the track length is 15 m. Each test track was driven with each control strategy, and each delay profile. Three repetitions for each configuration were carried out. The test run was declared successful if the robot reached the finish line without collision. The testing procedure included following steps:

- observe the operation of the system under manual, semiautonomous, and autonomous control strategies,
- 2. measure time, location, and collisions, and
- 3. measure the improvement (if any) of using autonomous or semi-autonomous system.



Fig. 11 The test tracks from left to right: 0, 1, 2, and 3

4.3 RGB-D Potential Field

To evaluate the performance of the presented methodology, we carry out visual analysis. Color and depth image-sets were recorded during a test run on the track 3. AF and RF were computed, which are presented in the image-set on Appendix 2. Herein, we present color images because they are more informative than the depth images.

Additionally, PF and its minimum are illustrated to visualize the input of the autonomous control function. Vertical black line indicates image center, the horizontal distance between the minimum and the line is proportional to y_{Lg} . Overall, method performed well as the AF minimum is aligned with the route line and obstacles are seen as peaks in the RF. The obstacle enlargement ensured that the minimum was shifted away from the obstacles.

Although the obstacles on the test tracks were cardboard boxes, we also tested the detection of different obstacles to study the robustness of the method. Figures 12 and 13 illustrate PF of variety of obstacles approximately at a distance of 1.5 m. The chair was detected correctly because the peak is as wide as the chair itself on the left of Fig. 12. The detection

of semi-transparent obstacle was not perfect, as the camera system sees partially through it, which is seen as non-uniform peak on the RF on the right of Fig. 12.

On the left of Fig. 13, the peak heights are relatively constant despite the obstacles had non-uniform height. This is because the peak height depends on the distance rather than the height of an obstacle. The right side on Fig. 13 illustrates that obstacle shorter than around 5 cm is not detectable with the current system. The reason for this is that the obstacle is between the virtual and the physical ground plane.

4.4 Trajectories

We analyze the behavior, performance, and limitations of the control strategies on the test tracks. Different control strategies produce different driving trajectories. As an example, Figure 14 illustrates typical trajectories of the Autonomous strategy. The position is illustrated once per second, so maximum distance between the dots is 0.5 m. The robot followed the route line well on the track 0. However, the trajectory deviated slightly from start line to 3 m due to initial heading. Besides that, deviation compared to the route line was negligible. The robot avoided obstacles and followed the route line successfully on tracks 1, 2, and 3. Most of the time, robot avoided obstacles with the PF method. CROA activated on the track 1 at 9.3 m, and multiple times on the track 3. Overall, the PF method performed better on tracks 1 and 2, where the density of obstacles was smaller compared to the track 3. However, no collisions happened during testing, which indicates that the control strategy is stable and robust.

Figure 15 presents typical trajectories of Delayed manual strategy. The color spectrum indicates the control delay. Trajectories indicate that there were lot of oscillations in the steering. Manual control has a benefit on the track 2, where the robot does not return on the route line after the first obstacle. Whereas Autonomous strategy does because it does not consider obstacles further than 3 m.

The typical trajectories of DDA strategy are illustrated on Fig. 16, where red indicates sections where the autopilot was enabled. Color spectrum indicates the sections driven with Delayed manual strategy and the control delay, and red indicates the sections driven with the autopilot.

Table 1Explanations of thecontrol strategies	Strategy	Explanation
-	Manual	Manual control without delay fluctuation
	Delayed manual	Manual control with delay fluctuation
	Autonomous	Autopilot is enabled during the whole run
	Delay-Dependent Assist (DDA)	Autopilot is enabled depending on the delay or if SZ=1
	Control-Dependent Assist (CDA)	Autopilot is enabled depending on the control signal difference or if SZ=1



Fig. 12 Illustration of the potential field on variety of obstacles



Fig. 13 Potential field of a collection of obstacles with different heights, and of an undetected short obstacle

Visual analysis of the trajectories indicate that the strategy works well, and the autopilot can take over rapidly depending on the delay. Autopilot was successful at detecting and returning on the route line while the trajectory deviated drastically from the route line. This is visible e.g. on the track 3 at 6 m and 11 m.

Lastly, the typical trajectories of CDA strategy are presented on Fig. 17. Again, red indicates the sections driven with the autopilot, which activated on all tracks. Autopilot



Fig. 14 Autonomous strategy: typical trajectories on the test tracks



Fig. 15 Delayed manual strategy (delay profile 1): typical trajectories on the test tracks

stabilized the oscillations, which were apparent with Delayed manual strategy. Autopilot helped maneuvering around the obstacles and was enabled significantly more on the track 3 compared to other tracks.

4.5 Control Performance

Control performance of the strategies was evaluated by measuring the time taken to complete the test tracks. Figure 18 presents arithmetic mean times with delay profiles, where gray indicates standard deviation. The absence of obstacles on the track 0 allowed the robot to move on a straight line, resulting in nearly constant completion times regardless on the control strategy. Overall, completion times on the track 3 were significantly larger compared to other tracks. Therefore, completion time is dependent on the density of obstacles. Figure 18 is analyzed from top to bottom, so first delay profile 1.

On tracks 1, 2, and 3, manual without delay was the quickest. Delayed manual was the slowest on tracks 1, 2, and 3. Autonomous, DDA, and CDA were quicker than Delayed manual on tracks 1, 2, and 3. DDA was slower compared to Autonomous on tracks 1 and 3, but quicker on the track 2. CDA was nearly as quick as manual without delay, especially on tracks 1 and 2. Completion times with delay profile 2 look similar to delay profile 1. On average, they are smaller compared to delay profile 1. DDA is quicker than Autonomous on tracks 1 and 3, whereas with delay profile 1, DDA was slower. Again, CDA was nearly as quick as manual without delay.

Based on the time spent on the most challenging track 3, it can be stated that CDA was the best, followed by Autonomous strategy, then DDA, with 12%, 9%, and 6% faster operation than Delayed manual strategy, respectively.



Fig. 16 DDA strategy (delay profile 1): typical trajectories on the test tracks. Red indicates the sections where autopilot was enabled



Fig. 17 CDA strategy (delay profile 1): typical trajectories on the test tracks. Red indicates the sections where autopilot was enabled

5 Discussion

5.1 RGB-D Potential Field

We noted that there are no valleys in the RF when the gaps between the obstacles are smaller than the width parameter w of the robot. Obstacles lower in the vertical axis than 5 cm were not detectable, because they are between the virtual and physical ground plane. The purpose of the gap between the planes is to prevent the physical ground plane intersecting the virtual ground plane. However, the implications of undetectable short obstacles are negligible because of the maneuverability of the robot. The problem could be solved



Fig. 18 The arithmetic mean track completion time, standard deviation in gray

by calibrating the virtual ground plane more accurately and mounting the camera more rigidly.

Reflective surfaces will certainly cause some problems for a vision-based system. Consequently, the performance of this system varies depending on the surface materials and the lighting conditions. For example, the reflective surfaces in the particle accelerator tunnel might cause false measurements with an RGB-D sensor. The same issue would be present in a LiDAR-based system too, while not detecting color, which is essential in the operation environment. Therefore, an RGB-D sensor is better suited for this application.

5.2 Trajectories

The RGB-DPF method avoided obstacles successfully on all test tracks. No collisions happened during the experiments with Autonomous strategy, which confirms the robustness of the proposed method. CROA activated more when the density of obstacles on route was high. We suspect that it activated because of the sudden appearance of the obstacles. The horizontal FOV of the camera system is relatively narrow (86 degrees at 2 m), which means that obstacles appear relatively late to the FOV in certain cases (*e.g.* on the track 2 at 9 m on Fig. 14). When this happens, the robot has already gotten so close to the obstacles that the CROA activates. The activation of CROA means that the intervention time increases because the robot briefly stops completely. We suspect that the overshoot on the track 1 at 10 m is caused by poor detection of the route line. The lighting on that part of the track was slightly dim, which made the detection of the line more unreliable.

Manual control with both delay profiles was difficult. The delay in the control signal caused oscillating steering. Controlling was more difficult when the speed of the robot was high. This indicates that assisted or autonomous control would be more useful on higher speeds.

DDA strategy prevented the "control-and-wait" style as the autopilot activated when the delay was over the threshold. However, it was challenging for the teleoperator to predict when the autopilot activates, which resulted in a decrease in operating confidence. CDA strategy provided the smoothest teleoperation experience. Teleoperating with this strategy was smooth because the autopilot was able to assist constantly unlike with DDA, and teleoperator could focus on the longitudinal control.

5.3 Control Performance

Both semi-autonomous strategies and Autonomous strategy made teleoperation smoother, which was seen as a decrease in track completion times. Completion times were highly dependent on the density of obstacles, as the times were short on the track 0 and long on the track 3. Time differences between control strategies were more evident with high density obstacles. Times on the track 0 were independent on the control strategy. It should be noted that the tracks were only 15 m long. To have better validation, tests on longer tracks must be conducted where time differences would become more apparent.

Autonomous strategy was quicker on the track 1 compared to the track 2 because on the track 2, the robot used time to maneuver on top of the route line between obstacles. We suspect that the completion times were collectively smaller with delay profile 2 because the teleoperator could send low delay control signals more frequently. Moreover, autopilot on DDA strategy activated more frequently resulting in better control.

In addition to perfect reliability, Autonomous strategy was also quicker than Manual strategy on the challenging track 3. The performance of the Autonomous strategy proves that autonomous navigation could be feasible also for long-term navigation, specifically in the particle accelerator tunnel because the environment is repetitive. Therefore, long-term path planning is not important in uniform, narrow, and one-directional tunnel if we assume that the robot has high accuracy localization system. Our system is computationally light, utilizing only a modest amount of processing resources, memory, and storage, as maps are not used nor long-term path planning performed.

6 Conclusion

The presented RGB-DPF method and the control strategies made the teleoperation smoother while the manual control signal had fluctuating delay. Based on the time spent on the most challenging track 3, it can be stated that CDA was the best, followed by Autonomous strategy, then DDA, with 12%, 9%, and 6% faster operation than Delayed manual strategy, respectively. The experiments proved that the completion time is dependent on the density of obstacles, and frequent low delay control signals make the teleoperation smoother compared to a case where large delay periods are longer. We demonstrated that during long cutouts (up to 5 s) the robot can be successfully navigated with a short-term autonomous control system, and teleoperator can gain the control back easily when the cutout ends. We developed an autonomous control system which is capable of navigating tunnel section, which mimics CERN particle accelerator tunnel, while utilizing only a modest amount of computation, memory, and storage as maps are not used nor long-term planning performed. Despite being designed for the accelerator tunnel; it can be easily applied to common industrial robots operating e.g. in warehouses or factories due to hardware simplicity and light computational demand.

Based on the results, the present study recommends using either control- or delay-dependent assist as a semi-autonomous strategy in presence of large delay fluctuation. While using a real robot prototype, these strategies were proven to perform well and reliably in challenging operation environments compared to a manual control strategy.

To conclude the limitations of this study, the usage of a single teleoperator incorporates subjectiveness to the results. Furthermore, the testing environment is not identical to the particle accelerator tunnel, especially in length. We mitigated the effect of this by conducting a large number (60) of repetitions in our testing environment. The final limitation is that the feedback was direct eyesight.

To address the discussed limitations, experiments with multiple teleoperators with a variety of skill levels should be carried out to increase the confidence of the results. Teleoperator feedback was direct eyesight, which is not the case in real operation scenarios as the robot is in a remote environment. Thus, video feedback from an on-board camera or from a remote environment should be tested.

Appendix



Fig. 19 The image-set from the on-board camera system and corresponding potential field during a run on the track 3

Image-set

The image-set is presented in Fig. 19.

Author Contributions Experimental set-up, planning, data collection, and analysis was performed by Alvari Seppänen. The first draft of the manuscript was written by Alvari Seppänen and Jari Vepsäläinen, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open access funding provided by Aalto University. Was provided by Helsinki Institute of Physics.

Data Availibility Statement The datasets generated during and analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflicts of interest The authors declare no conflict of interest.

Ethics approval was obtained from the ethics committee of Aalto University.

Consent for publication Consent was obtained from all individual participants included in the study. The authors has consented to the submission of the research manuscript to the journal.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

- Masmoudi, M., El Boudadi, L.K., Loukil, A., Dahane, A., Khelf, M.: The teleoperation of a mobile robot in a network without a quality of service guaranteed. In: 2015 12th International Symposium on Programming and Systems (ISPS), pp. 1–6 (2015). https://doi.org/10.1109/ISPS.2015.7244962
- Anderson, R.J., Spong, M.W.: Bilateral control of teleoperators with time delay. In: Proceedings of the 1988 IEEE International Conference on Systems, Man, and Cybernetics, vol. 1, pp. 131– 138 (1988). IEEE
- Lunghi, G., Marin, R., Di Castro, M., Masi, A., Sanz, P.J.: Multimodal human-robot interface for accessible remote robotic interventions in hazardous environments. IEEE Access 7, 127290– 127319 (2019). https://doi.org/10.1109/ACCESS.2019.2939493
- Held, R., Efstathiou, A., Greene, M.: Adaptation to displaced and delayed visual feedback from the hand. J. Exp. Psychol. 72(6), 887 (1966)

- You, B., Li, J., Ding, L., Xu, J., Li, W., Li, K., Gao, H.: Semiautonomous bilateral teleoperation of hexapod robot based on haptic force feedback. J. Intell. Robot. Syst. 91(3), 583–602 (2018)
- Liu, Y.-C., Chopra, N.: Control of semi-autonomous teleoperation system with time delays. Automatica 49(6), 1553–1565 (2013). https://doi.org/10.1016/j.automatica.2013.02.009
- Zhai, D.-H., Xia, Y.: Adaptive control of semi-autonomous teleoperation system with asymmetric time-varying delays and input uncertainties. IEEE Trans. Cybern. 47(11), 3621–3633 (2017). https://doi.org/10.1109/TCYB.2016.2573798
- Liu, Y.-C.: Task-space bilateral teleoperation systems for heterogeneous robots with time-varying delays. Robotica 33(10), 2065–2082 (2015)
- Li, W., Li, Z., Liu, Y., Ding, L., Wang, J., Gao, H., Deng, Z.: Semi-autonomous bilateral teleoperation of six-wheeled mobile robot on soft terrains. Mech. Syst. Signal Process. 133, 106234 (2019). https://doi.org/10.1016/j.ymssp.2019.07.015
- Cho, J.-H., Pae, D.-S., Lim, M.-T., Kang, T.-K.: A real-time obstacle avoidance method for autonomous vehicles using an obstacledependent gaussian potential field. J. Adv. Transp. 2018 (2018)
- Shen, S., Song, A., Li, T.: Bilateral motion prediction and control for teleoperation under long time-varying delays. ISA transactions 115, 61–70 (2021)
- Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. In: Autonomous Robot Vehicles, pp. 396–404. Springer, (1986)
- 13. Xu, T., Zhou, H., Tan, S., Li, Z., Ju, X., Peng, Y.: Mechanical arm obstacle avoidance path planning based on improved artificial potential field method. Industrial Robot: the international journal of robotics research and application (2021)
- Hein, B.: Collision avoidance for mobile robots using proximity sensors. In: Interactive Collaborative Robotics: 6th International Conference, ICR 2021, St. Petersburg, Russia, September 27–30, 2021, Proceedings, p. 205 (2021). Springer Nature
- Yao, Q., Zheng, Z., Qi, L., Yuan, H., Guo, X., Zhao, M., Liu, Z., Yang, T.: Path planning method with improved artificial potential field-a reinforcement learning perspective. IEEE Access 8, 135513–135523 (2020). https://doi.org/10.1109/ACCESS.2020. 3011211
- Orozco-Rosas, U., Picos, K., Montiel, O.: Hybrid path planning algorithm based on membrane pseudo-bacterial potential field for autonomous mobile robots. IEEE Access 7, 156787–156803 (2019). https://doi.org/10.1109/ACCESS.2019.2949835
- Rostami, S.M.H., Sangaiah, A.K., Wang, J., Liu, X.: Obstacle avoidance of mobile robots using modified artificial potential field algorithm. EURASIP J. Wirel. Commun. Netw. **2019**(1), 1–19 (2019)
- Park, S.-O., Lee, M.C., Kim, J.: Trajectory planning with collision avoidance for redundant robots using jacobian and artificial potential field-based real-time inverse kinematics. Int. J. Control Autom. Syst. 18(8), 2095–2107 (2020)
- Wen, G., Chen, C., Dou, H., Yang, H., Liu, C.: Formation control with obstacle avoidance of second-order multi-agent systems under directed communication topology. Sci. China Inf. Sci. 62(9), 1–14 (2019)
- Rasekhipour, Y., Khajepour, A., Chen, S.-K., Litkouhi, B.: A potential field-based model predictive path-planning controller for autonomous road vehicles. IEEE Trans. Intell. Transp. Syst. 18(5), 1255–1267 (2017). https://doi.org/10.1109/TITS.2016.2604240
- P.V.C., H.: Method and means for recognizing complex patterns (U.S. Patent 3,069,654, Dec. 18, 1962)
- 22. Coulter, R.C.: Implementation of the pure pursuit path tracking algorithm. Technical report, Carnegie-Mellon UNIV Pittsburgh PA Robotics INST (1992)

- 23. Moore, T., Stouch, D.: A generalized extended kalman filter implementation for the robot operating system. Proceedings of the 13th International Conference on Intelligent Autonomous Systems (IAS-13) (2014)
- Rizzo, C., Sicignano, D., Riazuelo, L., Tardioli, D., Lera, F., Villarroel, J.L., Montano, L.: Guaranteeing communication for robotic intervention in long tunnel scenarios. In: Reis, L.P., Moreira, A.P., Lima, P.U., Montano, L., Muñoz-Martinez, V. (eds.) Robot 2015: Second Iberian Robotics Conference, pp. 691–703. Springer, Cham (2016)
- Giger, A., Barnett, W.: Effects of multipath propagation on digital radio. IEEE Transactions on Communications 29(9), 1345–1352 (1981). https://doi.org/10.1109/TCOM.1981.1095148
- Stark, W.: Effects of multipath fading in wireless communication systems. Microwave and RF Product Applications, 22–1 (2003). https://doi.org/10.1201/9780203503744.ch22
- Di Castro, M., Ferre, M., Masi, A.: Cerntauro: A modular architecture for robotic inspection and telemanipulation in harsh and semi-structured environments. IEEE Access 6, 37506–37522 (2018). https://doi.org/10.1109/ACCESS.2018.2849572

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Alvari Seppänen received his BSc and MSc degrees from Aalto University in 2020 and 2021, respectively. He is currently studying towards a DSc degree at Aalto University in the Autonomy and Mobility laboratory group. He has been included in the Dean's list for academic years 2018–2019, 2019–2020, and 2020–2021 for exceptional study performance. Mobile robotics and robot perception are his main research interests.

Dr. Jari Vepsäläinen received BSc, MSc and DSc degrees at Aalto University in 2014, 2016 and 2019. During his bachelor and master level studies, he concurrently worked as a Research Assistant in the Fluid Power laboratory.Jari is currently continuing his academic career as a post-doctoral researcher at Aalto University since early 2020. He has a background in electric vehicles and uncertainty quantification. Dr. Vepsäläinen has also done work in various data science projects and around machine learning topics. During his doctoral studies, he was also involved in course planning and assisting in the teaching of mechatronics, which is currently his main responsibility.

Risto Ojala received his BSc and MSc degrees from Aalto University in 2019 and 2021, respectively. Currently, he continues his studies towards a DSc degree at Aalto University in the Autonomy and Mobility laboratory group. He has authored several peer-reviewed journal publications, and his research interests focus on smart vehicles, applied computer vision and machine learning.

Prof. Kari Tammi received MSc, LicSc, and DSc degrees at Helsinki University of Technology in 1999, 2003, and 2007. He received a Teacher's pedagogical qualification at Häme University of Applied Sciences in 2017. He has worked as an Associate Professor at Aalto University since 2015. He started as Vice Dean for Impact in 2022.

Kari also serves the Finnish Administrative Supreme Court as Chief Engineer Counselor. Earlier, he worked as Research Professor, Research Manager, Team Leader and in other positions at VTT Technical Research Centre of Finland (2000–2015). He has been a post-doctoral researcher at North Carolina State University (USA, 2007–2008), and a researcher at CERN, the European Organization for Nuclear Research in 1997–2000. He has authored more than 100 publications cited in more than 6000 other publications. Dr. Tammi currently serves as the deputy chair for IFTOMM Finland.