CoNav Chair: Design of a ROS-based Smart Wheelchair for Shared Control Navigation in the Built Environment

Yifan Xu[1], Qianwei Wang[2], Jordan Lillie[3], Vineet Kamat[4] and Carol Menassa[5]

[1]Department of Civil and Environmental Engineering, University of Michigan, United States of America
[2]College of Literature, Science, and the Arts, University of Michigan, United States of America
[3]Wheelchair Seating Service, University of Michigan Health, United States of America
[4]Department of Civil and Environmental Engineering, University of Michigan, United States of America
[5]Department of Civil and Environmental Engineering, University of Michigan, United States of America

yfx@umich.edu, qweiw@umich.edu, jlillie@med.umich.edu, vkamat@umich.edu, menassa@umich.edu

Abstract -
With the number of people with disabilities (PWD) increasing worldwide each year, the demand for mobility support to enable independent living and social integration is also growing. Wheelchairs commonly support the mobility of PWD in both indoor and outdoor environments. However, current powered wheelchairs (PWC) often fail to meet the needs of PWD, who may find it difficult to operate them. Moreover, existing research on robotic wheelchairs typically focuses either on full autonomy or enhanced manual control, which can lead to reduced efficiency and user trust. To address these issues, this paper proposes a Robot Operating System (ROS)-based smart wheelchair (CoNav Chair) that incorporates a shared control navigation algorithm and obstacle avoidance to support PWD while fostering efficiency and trust between the robot and the user. Our design is divided into hardware and software components. Experimental results conducted in a typical indoor social environment demonstrate the performance and effectiveness of our smart wheelchair's hardware and software design. This integrated design fosters trust and autonomy, crucial for the acceptance of assistive mobility technologies in the built environment.

Keywords -
Smart Wheelchair; Navigation; Shared Control

1 Introduction

With the number of people with disabilities (PWD) increasing annually, more and more people will have to rely on wheelchairs to facilitate their mobility to live their lives independently. In the United States, nearly 2.7 million people experience mobility challenges leading them to rely on wheelchairs in their daily lives [1]. That is expected to increase by 7% annually [2] leading to a growing demand for mobility support systems that enable independent living and social integration. Wheelchairs, especially powered wheelchairs (PWC), are the most common means of assisting PWD in both indoor and outdoor environments where these devices provide crucial mobility assistance, allowing users to navigate their surroundings more easily [3]. In addition, smart wheelchairs created to introduce autonomy in PWC operation have demonstrated significant potential in improving the quality of life for PWD [4].

Current smart wheelchairs face a notable limitation: they often focus exclusively on either full autonomy or enhanced manual control [4]. This imbalance can result in inefficiencies during navigation and a lack of trust in the technology—critical factors that influence the successful adoption of these devices [5]. Shared control enables users to partially override the autonomous navigation system, allowing them to contribute to decision-making during navigation. This approach not only enhances the mobility of PWD but also fosters user trust by offering a balanced partnership between autonomy and manual control.

Building on insights from previous smart wheelchair designs [6, 7, 8, 9, 10, 11] and shared control-based navigation structures [12], we developed both the hardware and software for such a shared control-based smart wheelchair that we call the CoNav Chair.

For hardware design, We based our hardware design on a widely used commercial PWC. The key elements include the integration of LiDAR, a camera, an inertial measurement unit (IMU), and two encoders. These sensors collectively provide comprehensive input data for the system. To accommodate all the sensors and the controller board, we designed a dedicated sensor mounting board located under the wheelchair. This sophisticated hardware integration lays the foundation for the implementation of the advanced shared control-based navigation algorithm, ultimately contributing to a more intuitive, reliable, and efficient mobility solution for people with disabilities. Additionally, we implemented a closed-loop proportional–integral–derivative (PID) [13] motor control module to manage the wheelchair's motors effectively.

The software design centers around a ROS-based Simultaneous Localization and Mapping (SLAM) and navigation framework, which facilitates navigation and interactive motor control. For the SLAM module, we employ Faster-LIO [14] to estimate the wheelchair's states and positions and RTAB-Map [15] to construct

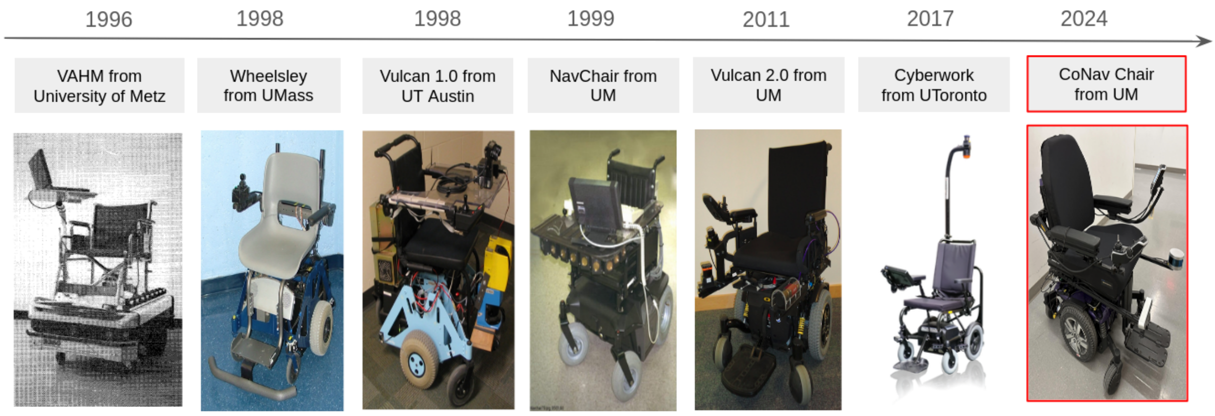

Figure 1. Development Timeline of Smart Wheelchairs

a detailed map of the indoor environment. This 2D map is then used in our shared control-based navigation system. By integrating the user's control input signals with the autonomous navigation system, the wheelchair can navigate effectively through the mapped area.

In general, our proposed system has the following contributions:

- We developed the CoNav Chair, a shared control-based smart wheelchair platform that bridges the gap between full autonomy and manual control by leveraging our proposed shared control algorithm, setting a new benchmark for assistive mobility solutions.
- At the hardware level, we developed a ROS-based smart wheelchair hardware platform built from a widely used commercial PWC with essential control electronics and sensors to enable advanced perception, mapping, autonomous navigation, and precise motor control.
- At the software level, we proposed a novel shared control-based navigation framework for assistive mobile robots integrating a user's desire for path preference and control during navigation to improve the efficiency and human-robot trust.
- We have tested our wheelchair system in a built environment and demonstrated that our robot system outperforms other autonomous wheelchair navigation systems both in effectiveness and user comfort.

2 Related Work

The development of smart wheelchairs has been a focal point in the literature aimed at improving the mobility and independence of PWD. A lot of research on wheelchair hardware instrumentation and software integration has been conducted and implemented to address the limitations of the traditional PWC.

The research on smart wheelchairs started in the 1990s, and a lot of work related to smart wheelchairs has been developed such as VAHM [6], Vulcan1.0 [7], Wheelsley [8] and NavChair [9], Vulcan2.0 [10] and Cyberwork Wheelchair [11]. The development timeline of the smart wheelchair system is shown in Figure 1. VAHM is equipped with a contact sensor, ultrasonic and infrared sensors to detect collisions, and wall-following algorithms to allow the wheelchair to navigate by following the wall. Wheelsley allows wheelchair users to give navigation commands by using an eye-tracking system. Vulcan1.0 uses 2D LiDAR and RGB camera input to build a visual 2D map for the wheelchair user to visualize their current location and their surroundings to let them better operate the wheelchair. The NavChair is an assistive wheelchair navigation system that integrates autonomous navigation, obstacle avoidance, door passage, and wall-following which allow the wheelchair to navigate autonomously. Vulcan2.0 is the improved version of Vulcan1.0 which integrates autonomous navigation algorithms and reduces the cost by using 2D LiDAR only. The Cyberwork Wheelchair is a commercially available smart wheelchair which is equipped with multiple RGB-D cameras to get the surrounding information and use it to navigate autonomously while avoiding obstacles. The performance comparison is shown in Table 1.

While these wheelchair platforms offer significant navigational assistance to individuals who struggle with traditional powered wheelchairs, they rely heavily either on sensor data or human input, which can have limitations and sensitivity to environmental changes [16]. For user controlled wheelchair systems, such as those using

Table 1. Comparison between the previous wheelchair models and the CoNav Chair

Wheelchair Models	SLAM compatible	Autonomous navigation	Shared control by user	Sensors
Powered wheelchair	N/A	N/A	N/A	N/A
VAHM [6]	N/A	N/A	N/A	Contact, Ultrasonic sensor
Wheelsley [8]	N/A	N/A	✓	Eye tracking system
Vulcan 1.0 [7]	✓	N/A	N/A	2D LiDAR and camera
Nav-Chair [9]	N/A	✓	N/A	12 ultrasonic sensors
Vulcan 2.0 [10]	✓	✓	N/A	2D LiDAR
Cyberwork [11]	✓	✓	N/A	3 Cameras and eye
Co-Nav Chair(Ours)	✓	✓	✓	3D LiDAR and RGB-D camera

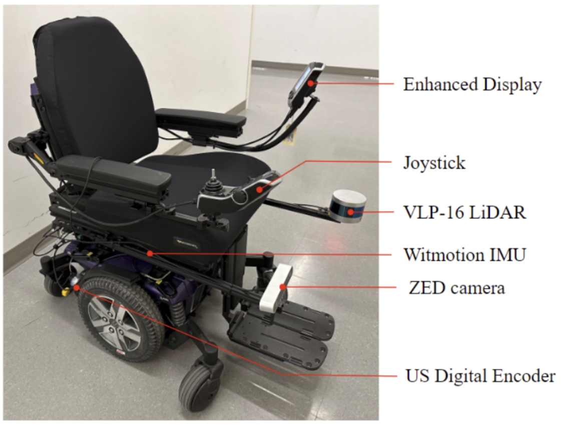

Figure 2. Hardware Design of the CoNav Chair

eye-tracking control [8], they require substantial cognitive effort or fine motor skills, which can be fatiguing or unreliable over extended periods. Fully autonomous systems, on the other hand, can fall short in dynamic and unpredictable environments due to sensor limitations or algorithmic shortcomings [17]. Our proposed system is equipped with multiple sensors and a shared control-based navigation system that allow users to partially control the wheelchair which increases the reliability of the intelligent wheelchair system, addressing the limitations of fully autonomous systems in dynamic and unpredictable environments.

3 Hardware Design

For this project, we have equipped a commercially available PWC, Quantum Q6 Edge 2.0, with control electronics, optical encoders, and multiple sensors such as VLP-16 LiDAR, ZED stereo camera, and Witmotion IMU. Although LiDAR and cameras are fundamental for these robotic functions, most commercial PWCs (e.g., Quantum Q6 Edge 2.0, Pride Mobility) are not manufactured with such sensing systems by default, and many current "smart" wheelchairs on the market remain prohibitively expensive or do not implement shared control. Inspired by the widely used Turtlebot3 platform [18], our goal is to develop a universal, research-oriented standard for shared-control smart wheelchairs. While our prototype employs a 3D LiDAR and an RGB-D camera to demonstrate robust performance, the architecture is modular—it can readily accommodate alternative sensors such as a 2D LiDAR or a single RGB-D camera. This flexible design allows researchers and manufacturers to balance cost and functionality as needed, thereby fostering accessible, shared-control mobility solutions for a broader range of users. The hardware architecture is shown in Figure 2.

As shown in Figure 3, Our hardware structure follows the Sensing - Reasoning - Acting structure. For the Sensing part, all of the sensors, such as the LiDAR and the camera will provide the robot with environmental information about the surroundings and publish the sensor data to a laptop PC for further processing. For the Reasoning module, the laptop PC is connected to the sensors and motors, which run a Linux operating system and ROS. Our shared control-based navigation program will run on this PC and send actuator control commands to motors. For the Acting module, a motor control system based on R-Net is designed, and a closed PID control system is integrated to ensure the stability of the lower-level control system.

3.1 Motion Control Module

For the motor control module, we are using the alternative control interface embedded in the wheelchair. The alternative control interface is on the enhanced display, which has a 9-way pin-out. We design our motor controller board, which can output two ways of voltage ranging from 4.8V - 7.2V to both control the forward and rotation velocity. The board is programmed to have a *rosserial* module to take command velocity signals sent from our laptop by publishing and subscribing nodes and topic protocol. Inside the board, two digital Potentiometers are used to convert the control message to the voltage signal and send it to the alternative control interface. After the alternative control interface gets the voltage output from the controller board, the interface will convert the voltage signals to CAN messages and send

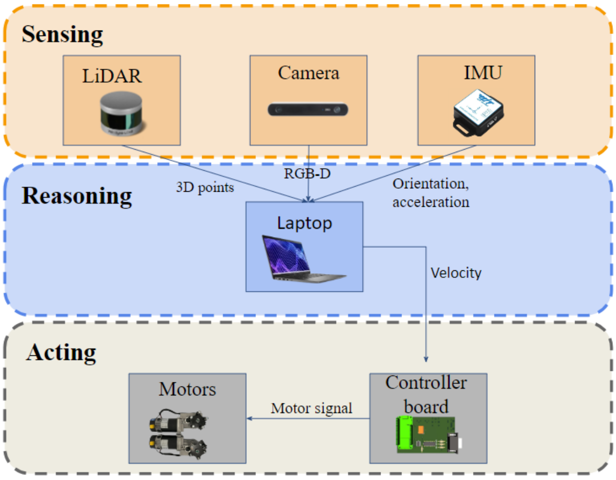

Figure 3. Hardware Dataflow of CoNav Chair

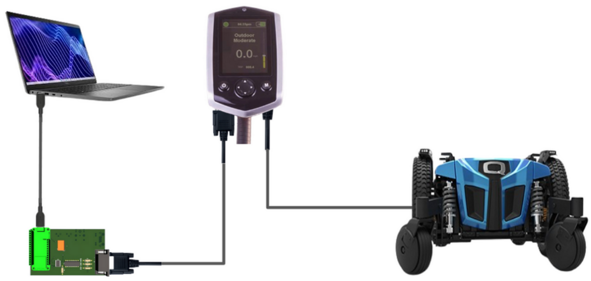

Figure 4. Motor Control Module

them to the wheelchair base. The way we connect the motor controller module is shown in Figure 4.

3.2 Sensor Module

The LiDAR and camera were mounted on the wheelchair using T-slotted aluminum extrusions to avoid obstructing leg movement. Both extrusions were attached to the wheelchair's armrest and seat rail using L brackets and square nuts, as shown in Figure 5. Cables were routed along the extrusions and secured with zip ties for a clean installation. The LiDAR and camera were mounted at the front of the wheelchair to prioritize forward-facing perception, since most powered wheelchair navigation involves moving and avoiding obstacles in the forward direction. Given that our wheelchair is a differential-drive system, it can rotate in place and thus rarely needs to travel backward; this reduces the necessity for rear coverage.

The wheelchair's motors were modified to retrofit new encoders while retaining the functionality of the integrated electromagnetic brakes. This involved preparing the motor shafts by drilling and threading them to accommodate an encoder mount. The encoders were securely installed

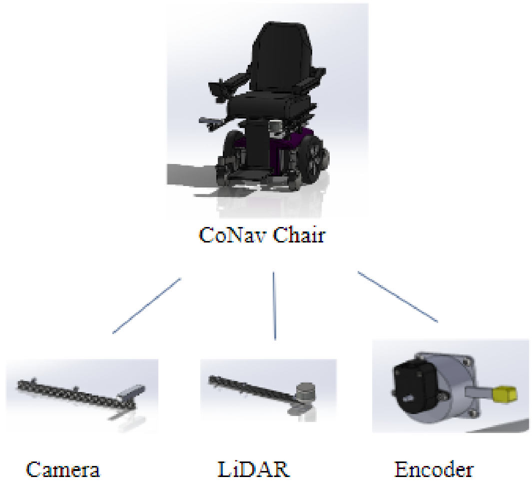

Figure 5. Sensor Module

using existing mounting points, following standard guidelines, to ensure precise and reliable operation.

4 Software Design

Our proposed SLAM and shared control-based navigation system of the smart wheelchair is based on ROS [19]. ROS provides a robust and flexible framework for writing robot software, allowing developers to leverage a vast ecosystem of tools, libraries, and conventions designed for robot applications. The general structure of the software arrangement is shown in Figure 6. The structure consists of three parts: a 3D mapping module, a Localization module, a Shared control-based navigation module, and a Motor control module.

4.1 Mapper and Localizer

For the mapping module, the system employs RTAB-Map [15], which is a graph-based SLAM algorithm capable of generating both 3D maps and their 2D projections of the surrounding environment. The RGB-D sensor serves as the primary input to RTAB-Map, enabling the construction of detailed 3D maps through real-time point cloud data processing. This map data is stored in a database, which is also used for managing obstacle point clouds for effective collision avoidance. The RTAB-Map framework integrates with a map server to publish the map information, accessible under the `/map` topic, which is crucial for high-level path planning and navigation tasks.

Simultaneously, the system utilizes Faster-LIO [14], a LiDAR-inertial odometry framework, for accurate localization. Faster-LIO processes data from the 3D LiDAR and IMU inputs. Initially, the raw LiDAR

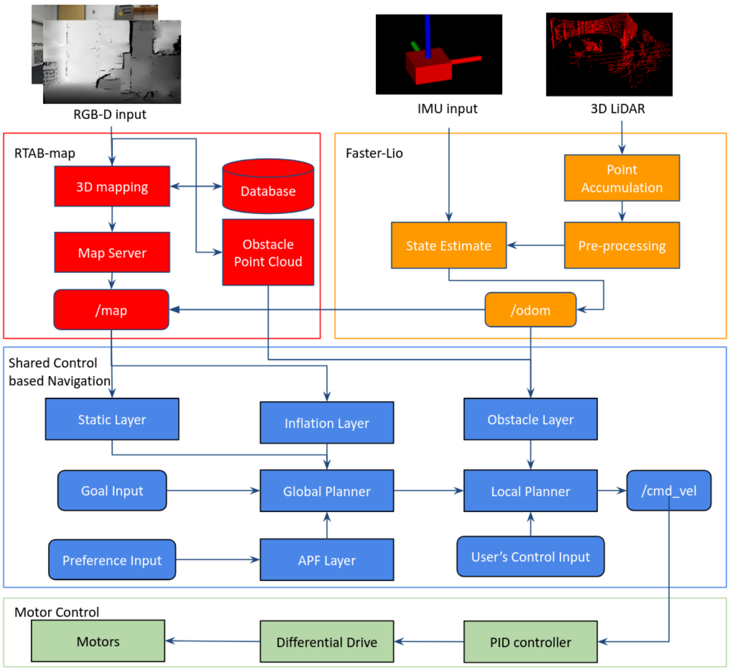

Figure 6. Software Structure of CoNav Chair

data undergoes pre-processing to filter out noise and structure the point cloud for efficient processing. These pre-processed point clouds are then accumulated to provide consistent spatial information. The state estimation module combines the IMU data with the processed LiDAR information, ensuring robust and precise localization even in dynamic environments. The resulting odometry data is published under the `/odom` topic, providing real-time position and orientation feedback to the wheelchair system.

4.2 Shared Control-based Navigation

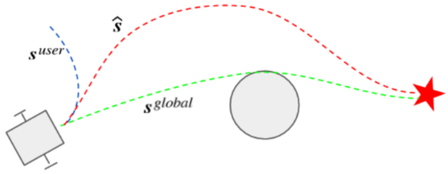

Figure 7. Shared Control-based MPC

Considering a wheelchair's role as an intimately user-integrated robotic system, it becomes essential to incorporate considerations of the user's control. Therefore, within our current framework, we have incorporated methods and concepts of shared control. In order to combine both user's control and autonomous navigation, we proposed a Shared Control-based Model Predictive Control local planner [12] to allow users to adaptively combine their control signals into the whole navigation process.

Here we utilized a non-linear model predictive control (MPC) [20] formulation for indoor environments navigation. The optimization problem is formulated below:

$$\begin{aligned} \boldsymbol{u}^* &= \arg\min_{\boldsymbol{u}\in\mathcal{U}} \mathcal{J}(\boldsymbol{s},\hat{\boldsymbol{s}},\boldsymbol{u}) \\ s.t. s_{t+1} &= f(s_t,u_t) \\ \boldsymbol{s} &\in \mathcal{S}_{free} \\ \boldsymbol{u} &\in \mathcal{U} \end{aligned} \tag{1}$$

where $s = \{s_0, ..., s_T\}$ is the state sequence of the robot drawn from the feasible set $\mathcal{S}_{free}$by passing a control sequence $u = \{u_0, ..., u_{T-1}\}$ drawn from a control space U through the robot dynamic f. $\hat{s} = \{\hat{s}_0, ..., \hat{s}_T\}$ is the combined reference states of robot by considering both the user's control signal sequence and the global path plan generated by the user-preference field. $\mathcal{J}$ is the cost function that expresses the considerations of efficiency, safety, and user control.

To enable the user to take control of the navigation process, we incorporate the predicted state generated by the user's control sequence into the cost function $\mathcal{J}$, as shown below.

$$\begin{aligned} \mathcal{J}(\boldsymbol{s}, \hat{s}, \boldsymbol{u}) &= \mathcal{J}_s(\boldsymbol{s}, \hat{s}) + \mathcal{J}_u(\boldsymbol{u}) \\ \mathcal{J}_s(\boldsymbol{s}, \hat{s}) &= (\boldsymbol{s} - \hat{\boldsymbol{s}})^T \boldsymbol{Q}_s (\boldsymbol{s} - \hat{\boldsymbol{s}}) \\ \hat{\boldsymbol{s}} &= \eta(k)\boldsymbol{s}^{user} + (1 - \eta(k))\boldsymbol{s}^{global} \\ \mathcal{J}^u &= \boldsymbol{u}^T \boldsymbol{Q}_u \boldsymbol{u} \end{aligned} \tag{2}$$

where $\mathcal{J}_s$ represents the state cost and $\mathcal{J}_u$ represents the input cost. The combined cost, denoted as $\hat{s}$, takes into account both the global planning state sequence $\boldsymbol{s}_{global}$ and the user's control sequence $\boldsymbol{s}_{user}$. The parameter $\theta(k) \in [0, 1]$ acts as a weight for the user's control sequence, determining the degree of control the user exerts. The variable k indicates the number of control signals the user provides within a given time frame. To adjust the levels of user control in our system, we devised a weight function based on the frequency of user control, reflecting their intent to steer the wheelchair. Specifically, each press of the user control interface (e.g., joystick or button) increases the weight of user input, indicating a stronger desire to steer the wheelchair. Conversely, if pressing frequency decreases, the system gradually shifts to rely more on autonomy. We capture this behavior with an exponential weighting function, as shown below, ensuring that urgent user interventions result in greater manual influence while non-urgent scenarios default to a higher degree of autonomous control. An exponential function is used to model the user's intent, as shown below:

$$\theta(k) = 1 - e^k \tag{3}$$

As shown in Figure 7, the reference path combined the user's control signal and the global path plan. The MPC local planner will then allow the robot to follow the combined path while avoiding obstacles.

5 Experiment Setup and Result

In this part, we set up a real-world experiment and tested our wheelchair within an indoor-built environment. The experiment setup and the result are shown below.

5.1 Experiment Setup

The CoNav Chair runs on Ubuntu 20.04 with ROS Noetic on board. The navigation task is set up in the corridor on the first floor of an academic building on our campus, shown in Figure 8. We created four corridor scenes, each within a 1.2m-wide hallway, where obstacles were distributed to test maneuverability. The CoNav Chair measures approximately 0.8m in width, leaving only 0.4 m of total clearance. The four scenes—Scene 1, Scene 2, Scene 3, and Scene 4—differ by the number of obstacles (3, 4, 5, and 6, respectively), introducing progressively tighter navigation constraints. One of these

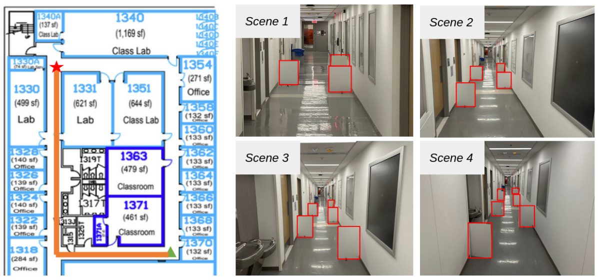

Figure 8. Experiment Environment and Navigation Route

scenes is shown in Figure 8. This setup was used to evaluate the performance of our shared-control algorithm against both fully autonomous and fully manual control methods. We assessed the planner based on time, success rate, collision rate, user satisfaction, trajectory length, and angle difference summary representing smoothness [21] against fully manual and autonomous controls.

5.2 Experiment Result

The quantitative results in Figure 9 show that the proposed shared control-based framework significantly outperforms fully manual and autonomous navigation [22] modes in both efficiency and reliability. The shared control planner achieves shorter completion times and trajectory lengths, reflecting faster and more direct navigation. Its smoother trajectories, indicated by lower cumulative absolute angle differences, ensure fewer abrupt turns and more natural directional changes—critical for applications like assistive robots where comfort and predictability are key. Additionally, the shared control planner demonstrates superior reliability, with higher success rates and fewer collisions across all scenarios.

From the qualitative results in Figure 10, we see that shared control yields fewer collisions compared to either autonomous or manual control. Under manual control, the trajectory is more curved, and users can lose control due to unfamiliarity with the system. In contrast, autonomous control tends to choose the most efficient path, sometimes compromising safety and resulting in collisions. With shared control, users can assume partial control as needed, allowing the system to safely avoid obstacles without risking a complete loss of control.

Manual control, while allowing users to choose direct routes, often suffers from increased collisions and lower success rates due to users' unfamiliarity with the robot's dynamics and excessive adjustments near obstacles. Autonomous navigation, on the other hand, excels at obstacle avoidance and path optimization but struggles in complex scenarios like sharp turns or centrally positioned obstacles, often leading to inefficiencies or 'freezing

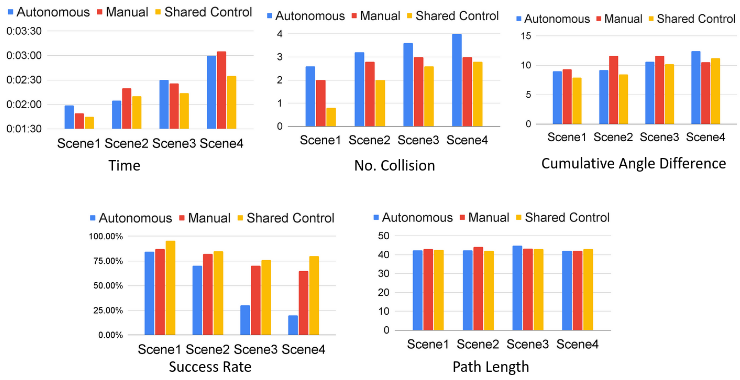

Figure 9. Quantitative Results of the CoNav Chair Evaluation Experiments

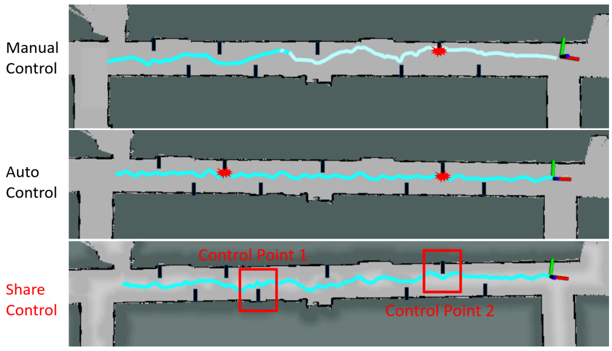

Figure 10. Qualitative Results of the CoNav Chair Evaluation Experiments. (Red explosion shape represents collision)

robot' issues [23]. The shared control framework combines the strengths of both approaches. It leverages autonomous navigation in simpler environments while enabling user intervention in complex scenarios, allowing for effective handling of challenging conditions. This hybrid strategy provides a balanced solution, enhancing efficiency, smoothness, and overall reliability.

6 Conclusion

In this work, we propose the CoNav Chair, a ROS-based smart wheelchair with shared control-based navigation and obstacle avoidance that addresses critical gaps in current PWC designs, by overcoming the limitations of fully autonomous or manual systems. Combining a multi-sensor hardware platform with LiDAR, a camera, an IMU, and encoders for real-time situational awareness and a ROS-based shared control software framework, our system integrates user inputs with autonomous navigation for smoother and more efficient mobility. Tested in real-world built environments, the CoNav Chair demonstrates superior performance in navigation effectiveness and user comfort, offering a reliable and intuitive solution to enhance independence for PWD.

7 Acknowledgement

The work presented in this paper was supported financially by the United States National Science Foundation (NSF) SCC-IRG 2124857. The support of the NSF is gratefully acknowledged.

References

[1] A. M. Koontz, D. Ding, Y. K. Jan, S. de Groot, and A. Hansen. Wheeled mobility. *BioMed Research International*, 2015:Article ID 138176, 2015. doi:10.1155/2015/138176.

[2] J. Flagg. Chapter 1: Wheeled mobility demographics. Technical report, Rehabilitation Engineering Research Center on Technology Transfer, University of New York at Buffalo, Buffalo, NY, 2009.

[3] R. A. Cooper. *Wheelchair Selection and Configuration*. Springer Publishing Company, 1998.

[4] L. Fehr, M. A. Langbein, and S. B. Skaar. Adequacy of power wheelchair control interfaces for persons with severe disabilities: A clinical survey. *Journal of Rehabilitation Research and Development*, 37(3): 353–360, 2000.

[5] Peter E McKenna, Marta Romeo, Jhielson Pimentel, Mohammed Diab, Meriam Moujahid, Helen Hastie, and Yiannis Demiris. Theory of mind and trust in human-robot navigation. In *Proceedings of the First International Symposium on Trustworthy Autonomous Systems*, TAS '23, page 1–5. ACM, July 2023. doi:10.1145/3597512.3597514. URL `http://dx.doi.org/10.1145/3597512.3597514`.

[6] G. Bourhis and P. Pino. Mobile robotics and mobility assistance for people with motor impairments: Rational justification for the vahm project. *IEEE Transactions on Rehabilitation Engineering*, 4(1): 7–12, 1996.

[7] W. S. Gribble, R. L. Browning, M. Hewett, E. Remolina, and B. J. Kuipers. Integrating vision and spatial reasoning for assistive navigation. In *Assistive Technology and Artificial Intelligence*, volume 1458 of *Lecture Notes in Computer Science*, pages 179–193. Springer, 1998. doi:10.1007/BFb0055978.

[8] H. A. Yanco. Wheelesley: A robotic wheelchair system: Indoor navigation and user interface. In *Assistive Technology and Artificial Intelligence*, volume 1458 of *Lecture Notes in Computer Science*, pages 256–268. Springer, 1998. doi:10.1007/BFb0055992.

[9] S. Levine, D. Bell, L. Jaros, R. Simpson, Y. Koren, and J. Borenstein. The NavChair assistive wheelchair navigation system. *IEEE Transactions on Rehabilitation Engineering*, 7(4):443–451, 1999.

[10] J. J. Park, S. Lee, and B. Kuipers. Discrete-time dynamic modeling and calibration of differential-drive mobile robots with friction. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6510–6517, 2017.

[11] F. Kopun. Toronto researchers develop technology for self-driving wheelchairs. *The Toronto Star*, 2023.

[12] Yifan Xu, Qianwei Wang, Vineet Kamat, and Carol Menassa. Socially-aware shared control navigation for assistive mobile robots in the built environment, 2024. URL `https://arxiv.org/abs/2405.17279`.

[13] Kiam Heong Ang, G. Chong, and Yun Li. Pid control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*, 13(4):559–576, July 2005. ISSN 1063-6536. doi:10.1109/tcst.2005.847331. URL `http://dx.doi.org/10.1109/TCST.2005.847331`.

[14] C. Bai, T. Xiao, Y. Chen, H. Wang, F. Zhang, and X. Gao. Faster-lio: Lightweight tightly coupled lidar-inertial odometry using parallel sparse incremental voxels. *IEEE Robotics and Automation Letters*, 7(2):4861–4868, 2022. doi:10.1109/LRA.2022.3152830.

[15] M. Labbé and F. Michaud. Appearance-based loop closure detection for online large-scale and long-term operation. *IEEE Transactions on Robotics*, 29(3): 734–745, 2013. doi:10.1109/TRO.2013.2242375.

[16] Jesse F. Leaman, Hung Manh La, and Luan Van Nguyen. Development of a smart wheelchair for people with disabilities. *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 279–284, 2016. URL `https://api.semanticscholar.org/CorpusID:5651927`.

[17] Sushil Kumar Sahoo and Bibhuti Bhusan Choudhury. Design and implementation of advanced sensor systems for smart robotic wheelchairs: A review. *Management Science Letters*, 14(3):151–168, 2024. ISSN 1923-9343. doi:10.5267/j.msl.2024.1.002. URL `http://dx.doi.org/10.5267/j.msl.2024.1.002`.

[18] Robin Amsters and Peter Slaets. Turtlebot 3 as a robotics education platform. In *Intelligent Robotics and Applications*, volume 11741 of *Lecture Notes in Computer Science*, pages 171–179. Springer, 2020. doi:10.1007/978-3-030-26945-6_16.

[19] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng. ROS: An open-source robot operating system. In *Proceedings of the ICRA Workshop on Open Source Software*, volume 3, 2009.

[20] A. Wang, C. I. Mavrogiannis, and A. Steinfeld. Group-based motion prediction for navigation in crowded environments. *arXiv preprint arXiv:2107.11637*, 2021. URL `https://arxiv.org/abs/2107.11637`.

[21] Silvia Guillén Ruiz, Luis V. Calderita, Alejandro Hidalgo-Paniagua, and Juan P. Bandera Rubio. Measuring smoothness as a factor for efficient and socially accepted robot motion. *Sensors*, 20(23), 2020. ISSN 1424-8220. doi:10.3390/s20236822.

[22] Amir Salimi Lafmejani and Spring Berman. Nonlinear mpc for collision-free and deadlock-free navigation of multiple nonholonomic mobile robots. *Robotics and Autonomous Systems*, 141:103774, July 2021. ISSN 0921-8890. doi:10.1016/j.robot.2021.103774. URL `http://dx.doi.org/10.1016/j.robot.2021.103774`.

[23] Peter Trautman and Andreas Krause. Unfreezing the robot: Navigation in dense, interacting crowds. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 797–803, 2010. doi:10.1109/IROS.2010.5654369.