We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

BOOK CITATION INDEX
CLARIVATE ANALYTICS
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Mobility Assistive Robots for Disabled Patients

Chung-Hsien Kuo
Department of Electrical Engineering
National Taiwan University of Science and Technology
Taiwan

1. Introduction

2.1 Background

Wheelchairs are one of the most important mobility aids for disabled patients and elders. In general, wheelchairs are categorized as manual and powered types. From the viewpoints of movement behaviours, the powered wheelchair can be regarded as a mobile robot (Asai, 2001) if additional environmental sensing components are built and an intelligent motion controller is included. Therefore, robotic wheelchairs (Luo et al., 1999) are proposed integrating novel sensing and intelligent control technologies to improve the convenience and comfort of mobility assistance.

In general, robotic wheelchairs integrate the sensing, intelligent computing, and communication technologies to improve the autonomy when compared to the conventional powered wheelchairs. The assistive technologies resolve the problems of conventional powered wheelchairs of:

1. Stable velocity control of wheels in terms of feedback motion control system.
2. Comfortable wheelchair driving in terms of velocity adjustment of wheels for various degrees of turning.
3. Collision avoidance of wheelchairs for approaching still and moving objects or person.

2.2 State of the art

In recent years, several robotic wheelchair projects and literatures were proposed. Their system focused on the autonomy, intelligence, safety, and navigation of robotic wheelchairs. The VAHM (French acronym for Autonomous Vehicle for people with Motor Disabilities) project proposed an autonomous wheelchair to assist the disabled people who are unpractical to drive a conventional powered wheelchair (Bourhis et al., 2001). The software architecture of VAHM robotic wheelchair is categorized as the physical, local and global levels. In the local level, the freespace detection, wall detection, and localization are implemented as the perception controller; the freespace search, direction following, wall following, path planning, motion control, obstacle avoidance are implemented as the navigation controller; and finally, the man-machine interface and command interpretation are implemented as the communication controller. The VAHM robotic wheelchair photo is shown in Fig. 1.

The SENARIO (Sensor Aided Intelligent Wheelchair Navigation System) project developed an intelligent wheelchair navigation system (Bourhis et al., 1997), and it provided fully

autonomous and semi-autonomous control modes. In the fully autonomous mode, the control system is capable of accepting the user's command. When the control system receives a "go to goal" command, it locates its current position and the target position from the map. Consequently, a path is generated, and the wheelchair executes by following the desired path. In addition, the wheelchair can avoid obstacles in terms of the ultrasonic and infrared sensors. In the semi-autonomous mode, the user's command can override the action generating from the autonomous control mode, and the users can drive the wheelchair using a specified path in a special environment. The SENARIO robotic wheelchair photo is shown in Fig. 1.

Fig. 1. Photo of robotic wheelchair for the VAHM project (Bourhis et al., 2001)

Fig. 2. Photo of robotic wheelchair for the SENARIO project (Bourhis et al., 1997)

The Bremen Autonomous Wheelchair project (Lankenau et al., 2001) developed a "Rolland" wheelchair, and it is implemented by following the skills of the driving and route assistants. The "Rolland" is capable of smooth speed control, turning around in an open space, turning around in a corner, obstacle avoidance, and doorway passage. It provides the mobility assistance for the handicapped people. The robotic wheelchair photo of 'Rolland' is shown in Fig. 3.

Fig. 3. Photo of Bremen autonomous wheelchair (Lankenau et al., 2001)

At the same time, the SIAMO (Spanish acronym for Integral System for Assisted Mobility) project (Mazo et al., 2001) developed an electronic system to guide the autonomous wheelchairs for the disabled and elder people. The proposed autonomous wheelchair system integrates an innovative user-machine interface, a complete sensory subsystem (ultrasonic, infrared, vision, etc), and an advanced strategy of control and navigation system to guide the wheelchair with safety and comfort. Especially, the face and mouth positions can be traced properly. The SIAMO robotic wheelchair photo is shown in Fig. 4.

Fig. 4. Photo of robotic wheelchair for the SIAMO project (Mazo et al., 2001)

The autonomous wheelchair can be also developed as the omni-wheeled platform. Such a driving platform is not suitable for the narrow space with non-straight paths of indoor environments, such as patient rooms in hospitals. In that paper, an omni-wheeled platform (Kuo et al., 2006) is proposed as the robotic wheelchair platform. Due to the property of free movements in orientations, the robotic wheelchair can easily drive in a narrow space with non-straight paths. In addition to the omni-wheeled platform, the robotic wheelchair also

integrates obstacle detection sensors to perform autonomous navigations using the fuzzy control system. However, such an omni-wheeled platform can be used in the indoor environment. A heavy-duty omni-wheel with larger size and a four-wheel configuration can be further used to improve the mechanical reliabilities. The omni-wheeled robotic wheelchair photo is shown in Fig. 5.
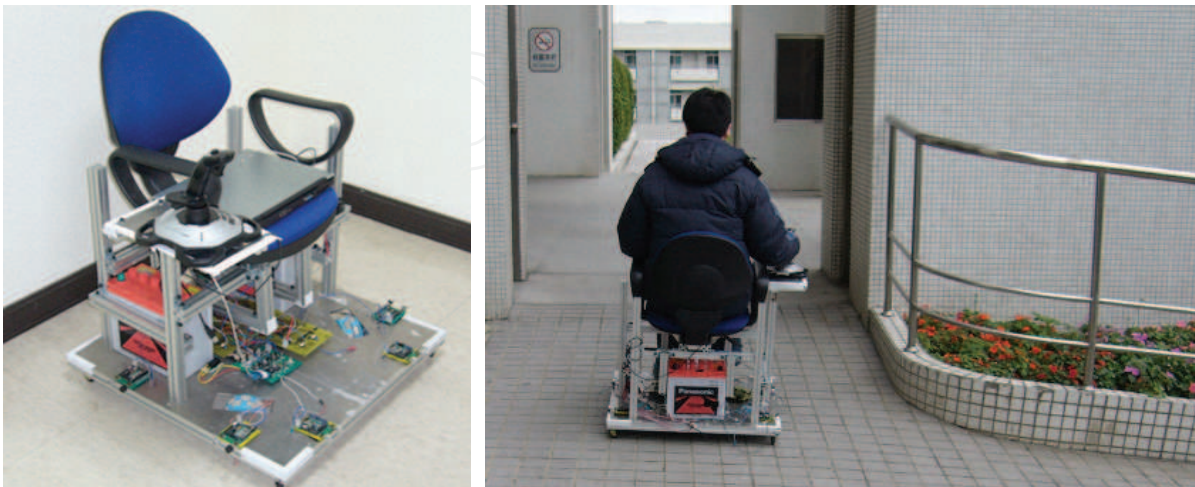

Fig. 5. Photo of omni-wheeled robotic wheelchair (Kuo et al., 2006)

2.3 Practical considerations of mobility assistive robots

Development of autonomous navigation system for robotic wheelchairs seems to be an important issue. However, most of navigation systems are developed based on personal computers such as desktops or laptops. Such a computational architecture increases the weight, cost and power consumption of the robotic wheelchair. Therefore, a chip based solution is discussed in this chapter to improve the feasibility of the robotic wheelchairs. Two important issues are proposed:

1. Human-centered design for mobility assistive robots: the conventional robotic wheelchair constructs the autonomous navigation technologies, and the user may drive the robotic wheelchair without using his (her) hand. Such an autonomous operation is not feasible when complicated environment presents. Therefore, the user joystick input should be reserved, and the joystick input can be regarded as a 'virtual' goal for the autonomous navigations. In this manner, the user is capable of controlling the wheelchair, and the intelligent robotic wheelchair controller is used to justify the joystick command for the safety and comfort considerations.
2. Low cost implementations: The cost for developing an autonomous wheelchair is quite expensive when compared to conventional powered wheelchair. The additional costs are ultrasonic sensors and personal computers. To reduce the cost, a distributed chip based robotic wheelchair supervisory controller is discussed in this work to improve the reliability, and to reduce the cost as well.

Finally, this chapter is organized as follows: section 3 introduces the fuzzy based navigations; section 4 describes the human-centered navigations; section 5 illustrates the distributed chip based implementations; section 6 illustrates the experiments and verifications; and finally the conclusions and future works are discussed in section 7.

3. Fuzzy based navigations
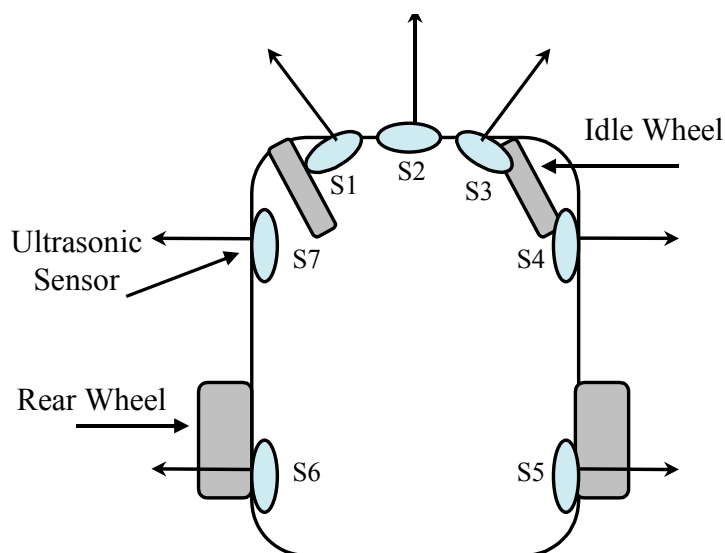
3.1 Fuzzy control system

The fuzzy control system developed for the robotic wheelchair is to emulate a skilled powered wheelchair user. The skilled powered wheelchair user drives the wheelchair depending on the following situations:

1. Goal direction and distance: the user drives the wheelchair in a higher velocity when the distance to the goal is long. The user reduces the velocity when the wheelchair rotates in a small radius for the safety and comfort concerns.
2. The user changes the wheelchair direction smoothly when obstacles appear in front of the wheelchair.
3. The user keeps a distance to the wall when a wall is beside the wheelchair.

According to the previous situations, the driving behaviors of a skilled robotic wheelchair user can be modeled as an intelligent controller. The intelligent controller percepts distance information of obstacles, and makes decision for the wheelchair steering and velocity. Therefore, the intelligent controller is responsible of three navigation functions, including wall-following, goal-seeking, and obstacle avoidance.

The fuzzy control system (Negnevitsky, 2005) uses the fuzzy sets and the membership functions to describe the facts with uncertainty. The inference engine is executed according to the input status and knowledgebase to determine the control policy. In this chapter, the robotic wheelchair uses the fuzzy logics as the kernel of the intelligent controller.

The case study in this chapter uses 7 ultrasonic sensors, and they are used as the inputs of the fuzzy controller, as shown in Fig. 6. The operation of the navigation fusion is indicated as in Fig. 7. In this figure, three navigation functions are dynamically switching according to the distributions of close obstacles. The switching reference table is shown in Table 1. In this table, the u1 to u7 indicate the signals collected form the ultrasonic sensors. Note that 'X' indicates no obstacle detected; 'O' indicates an obstacle is detected; and '-' indicates 'don't care'. In addition, the steering (i.e., velocity allocations of two wheels) control policy can be inferred from the results of the wall-following, goal-seeking, and obstacle avoidance fuzzy functions.

Si indicates the ultrasonic sensor *i*
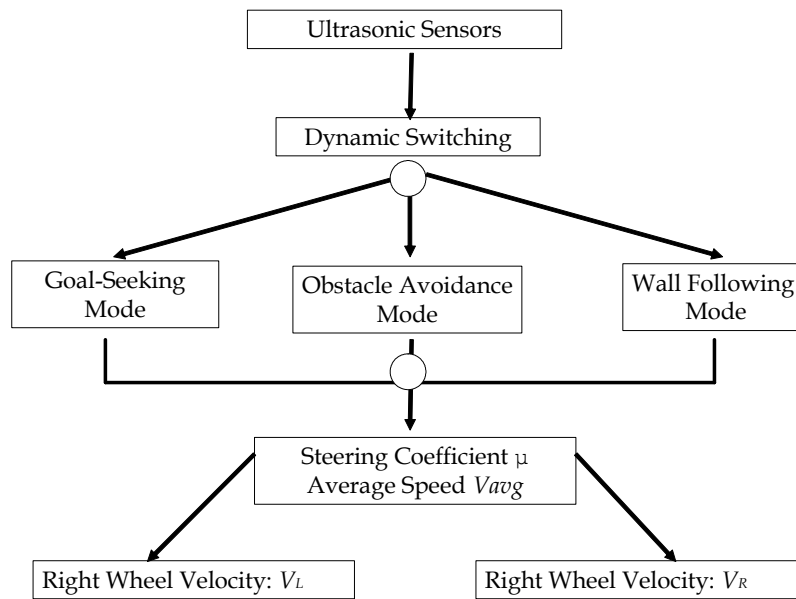
Fig. 6. Sensor allocations of case study

Fig. 7. Fusions of navigation functions

u1	u2	u3	u4	u5	u6	u7	Execution Function
×	×	×	×	×	×	×	Goal-Seeking
×	×	×	×	○	○	×	Goal-Seeking
×	×	×	×	○	×	×	Goal-Seeking
×	×	×	×	×	○	×	Goal-Seeking
×	×	×	○	○	×	×	Wall-Following
×	×	×	×	×	○	○	Wall-Following
×	×	×	○	○	—	○	Wall-Following
×	×	×	○	○	○	—	Wall-Following
×	×	×	—	○	○	○	Wall-Following
×	×	×	○	—	○	○	Wall-Following
○	—	—	—	—	—	—	Obstacle Avoidance
—	○	—	—	—	—	—	Obstacle Avoidance
—	—	○	—	—	—	—	Obstacle Avoidance
○	○	—	—	—	—	—	Obstacle Avoidance
○	—	○	—	—	—	—	Obstacle Avoidance
—	○	○	—	—	—	—	Obstacle Avoidance
○	○	○	—	—	—	—	Obstacle Avoidance

Table 1. Navigation fusion table

3.2 Goal-seeking mode

The goal-seeking mode is defined as none of the ultrasonic sensor detected obstacles, and the sensors in the same side of the robot ('S4 and S5' or 'S6 and S7') do not detect any obstacles simultaneously (as indicated in Table 1). For this navigation mode, the fuzzy rule table is shown in Table 2. Where θ is the angle between the direction of the wheelchair (α) and the direction of from two rear wheels' centre to the target (β), as shown in Fig. 8; μ is the steering coefficient. The linguistic parameters of fuzzy control are also presented: NB indicates negative big; NM indicates negative medium; NS indicate negative small; ZE indicates zero; PS indicates positive small; PM indicates positive medium; and PB indicates positive big. The input membership function of θ and the output membership function of μ are shown in Fig. 9(a) and Fig. 9(b), respectively.
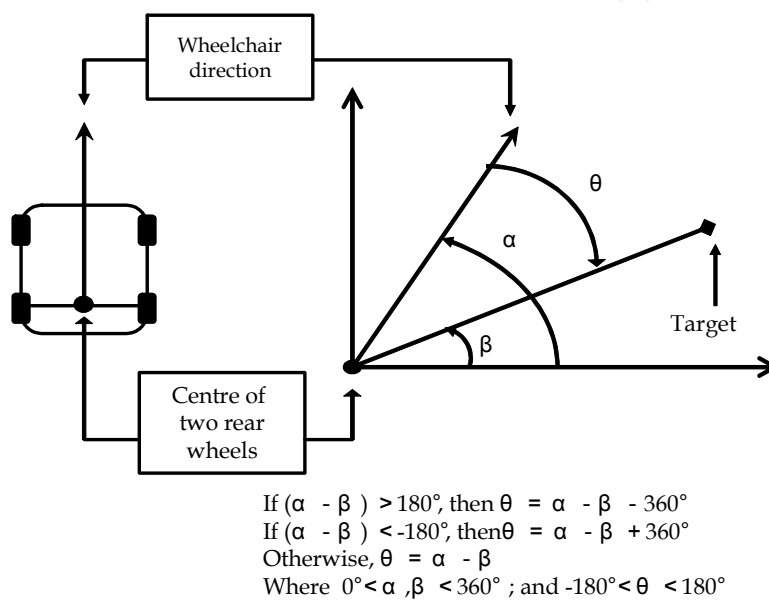

If $(\alpha - \beta) > 180°$, then $\theta = \alpha - \beta - 360°$
If $(\alpha - \beta) < -180°$, then $\theta = \alpha - \beta + 360°$
Otherwise, $\theta = \alpha - \beta$
Where $0° < \alpha, \beta < 360°$; and $-180° < \theta < 180°$

Fig. 8. Angle parameter definition for fuzzy controller

Fig. 9. Goal-seeking mode membership functions

θ	NB	NM	NS	ZE	PS	PM	PB
μ	NB	NM	NS	ZE	PS	PM	PB

Table 2. Goal-seeking mode fuzzy rule table

3.3 Wall-following mode

The wall-following mode is defined as S1, S2, and S3 which do not detect any obstacles and two ultrasonic sensor pair in the same side (S4, S5 pair or S6, S7 pair) which detects obstacles. In this mode, the fuzzy rule table is shown in Table 3. Their membership functions are shown in Fig. 10.

Condition 1: following the right wall

μ		$Ds4$			
		S	M	B	L
$Ds5$	S	NS	PS	PM	PM
	M	NS	ZE	PS	PM
	B	NM	NS	PS	PM
	L	NM	NM	ZE	PM

Condition 2: following the left wall

μ		$Ds7$			
		S	M	B	L
$Ds6$	S	PS	NS	NM	NM
	M	PS	ZE	NS	NM
	B	PM	PS	NS	NM
	L	PM	PM	ZE	NM

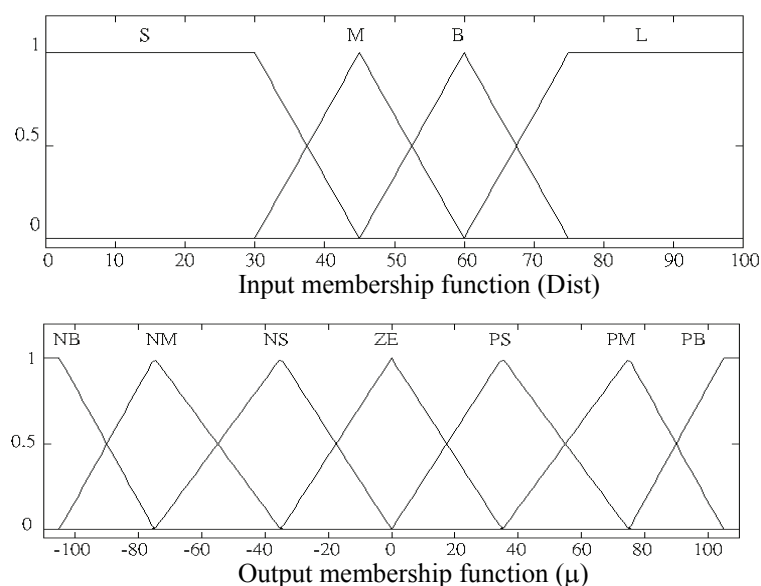Table 3. Wall-following mode fuzzy rule table

Fig. 10. Wall-following mode membership functions

3.4 Obstacle avoidance mode

The obstacle avoidance mode is defined as at least one of the front ultrasonic sensors (S1, S2, or S3) which detect obstacles. In this mode, the fuzzy rule table is defined in Table 3. Their membership functions are shown in Fig. 11. Where Dsi is the distance measured from Si.

Condition 1: if Ds1 is Small (S)

μ		$Ds3$			
		S	M	B	L
$Ds2$	S	PB	PB	PB	PB
	M	PB	PB	PB	PB
	B	PB	PB	PB	PB
	L	PB	PB	PB	PB

Condition 2: if Ds1 is Medium (M)

μ		$Ds3$			
		S	M	B	L
$Ds2$	S	PB	PB	PB	PB
	M	NB	PB	PB	PB
	B	NB	PB	PM	PM
	L	NB	PB	PM	PM

Condition 3: if Ds1 is Big (B)

μ		$Ds3$			
		S	M	B	L
$Ds2$	S	NB	NB	PB	PB
	M	NB	NB	PB	PB
	B	NB	NM	PM	PM
	L	NB	NM	PM	PS

Condition 4: if Ds1 is Large (L)

μ		$Ds3$			
		S	M	B	L
$Ds2$	S	NB	NB	NB	PB
	M	NB	NB	NB	PB
	B	NB	NM	NM	PM
	L	NB	NM	NS	ZE

Table 4. Obstacle avoidance mode fuzzy rule table

3.5 Velocity justification model

The navigation modes are executed in a mutual exclusion manner. The output of navigation mode is the steering coefficient, and it determines the degree of turning of the robotic wheelchair. Subsequently, the steering coefficient and the minimal distance of from wheelchair to desired target or nearest obstacle (denoted as dist) are further used to determine the average driving velocity (denoted as Vavg) of the robotic wheelchair. Such a

design aims to increase the safety of steering, and it is quite important to avoid the turnover of wheelchairs from their sides. In this manner, increasing steering angle will result in the decreasing in speed of the wheelchair. Such a design meets the behaviors of skill wheelchair user mentioned at the beginning of this section. Consequently, the fuzzy rule table is shown in Table 5. Their membership functions are shown in Fig. 12.
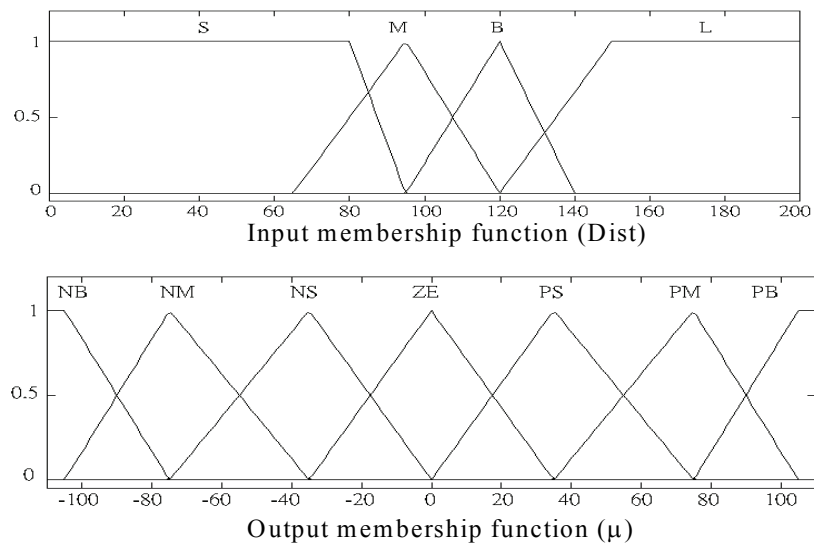

Fig. 11. Obstacle avoidance mode membership functions

V$_{avg.}$		dist			
		S	M	B	L
μ	NB	ZE	ZE	PS	PM
	NM	ZE	ZE	PS	PM
	NS	ZE	PS	PM	PB
	ZE	ZE	PS	PM	PB
	PS	ZE	PS	PM	PB
	PM	ZE	ZE	PS	PM
	PB	ZE	ZE	PS	PM

Table 5. Velocity justification fuzzy rule table

4. Human-centered navigations

Most of conventional robotic wheelchairs construct their intelligent navigation system using autonomous manners. However, such an autonomous operation is not feasible when in complicated environments. Therefore, the user joystick input should also be the best way to control the robotic wheelchair. Nevertheless, the joystick may be sensitive to user wrist input. Improper inputs may result in the dangers of users. The proposed fuzzy based navigation functions should be included to avoid dangerous situations and to improve safety and comfort before the wheel control command is made.

From previous discussions, the fuzzy navigation model can be properly executed only when the target position is determined. For most of users, they have no idea about how to set the target positions. It is also inconvenient to force users to set the target positions for each

segmental path. More seriously, the global localization and map of environment is frequently not available for the robotic wheelchairs for most situations. Consequently, the autonomous navigation technology is not feasible for mobility assistances.
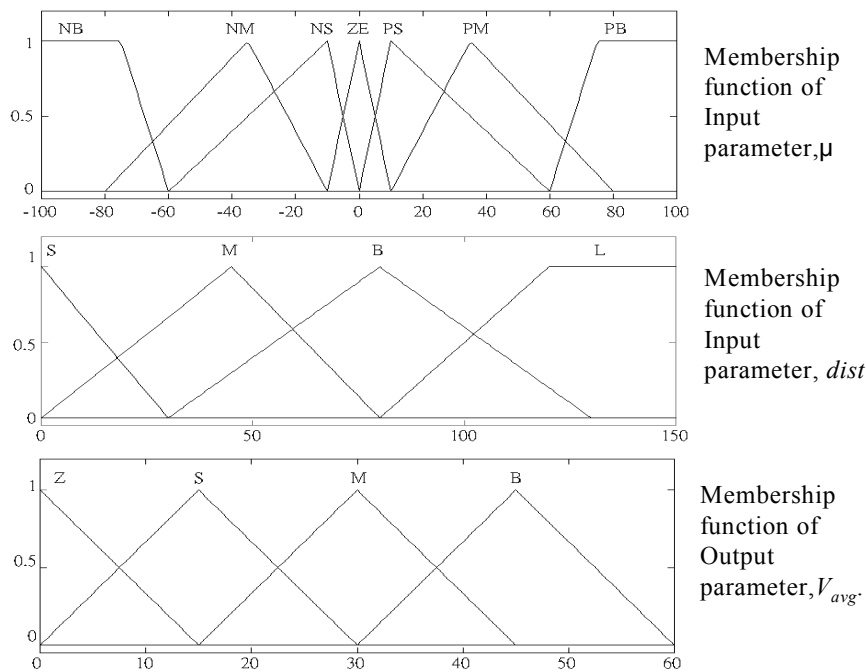

Fig. 12. Velocity justification model membership functions

In this work, the intelligent navigation technology will be constructed without the global positioning and electric environment map. More specially, no target position is required for the robotic wheelchair. The wheelchair users can use their eyes to understand where they are and to use the joystick to determine where to go. The navigation functions are used to assist the user to control the robotic wheelchair in more safe and comfortable.

Detailed description of the human-centered navigation is shown in Fig. 13. The joystick command will be normalized as (pos_x, pos_y). To determine the target of the navigation functions, the normalized joystick command will be further transformed as the virtual target position. In order to improve the comfortability and to reduce the sensitivity of the joystick, the joystick command will not be converted linearly.

Instead, a hybrid function is proposed by combining exponential and parabolic functions, as shown in Fig. 14 . The hybrid function Y is defined as:

$$Y = e^{mw} - 1 \qquad (w <= 0.5\, Jmax) \qquad (1)$$

$$Y = aw^2 + bw \qquad (w > 0.5\, Jmax) \qquad (2)$$

Where Y is the output of the hybrid function; w is smoothed signal of the joystick; a, b and m are constants. Note that Y can be used to determine target_x and target_y independently. Jmax is the maximum value of normalized pos_x or pos_y (i.e., unity). Consequently,

$$a = \left(2Y_{max} - 4Y_{set}\right) / J_{max}^2 \qquad (3)$$

$$b = \left(4Y_{set} - Y_{max}\right) / J_{max} \qquad (4)$$

$$m = 2\ell n(Y_{set} + 1) / J_{max} \tag{5}$$

Where Ymax is the maximum meaningful distance to the target for x-and y-coordinates of the robotic wheelchair; and Yset is an adjustable parameter, and it is determined in terms of the health condition of the user. For example, the Ymax is set as 400 cm, and this setting indicates that target position in a distance (either x or y direction) larger than 400 cm should conduct similar results. Such a value can be regarded as the limit position of joystick in each direction.

Fig. 13. Detailed description of the human-centered navigation

Fig. 14. Hybrid function conversion for joystick input

The target position cooperating with the ultrasonic sensor data will be used for the fuzzy navigation function to generate the steering coefficient (TC). Finally, the TC and the minimum distance of obstacle or the target are used to infer the angular velocities of two wheels.

5. Distributed chip based implementations

As mention before, most of conventional robotic wheelchair control systems are constructed using the personal computers, and they are less reliable. At the same time, the cost and weight are also drawbacks. In this work, a distributed chip based robotic wheelchair supervisory controller is discussed. For this purpose, a low cost computing processor of the programmable system-on-chip (PSoC) (http://www.cypress.com, 2008) is selected. In general, the performance of low cost PSoC is limited.

To extend computational capability of the PSoC, a distributed computing architecture is proposed. The individual navigation function are implemented in a task based PSoC. Consequently, the operations of the whole system achieve acceptable performance in real-time. The proposed distributed computing architecture is shown in Fig. 15. The intra-communications are established in terms of the I2C and the serial interfaces. There are 10 PSoC modules used in this work, and they are marked using the gray rectangles. These modules are further elaborated as follows:

Fig. 15. Distributed PSoC based computing architecture

1. Joystick input/ display module: this module is responsible of collecting user's joystick command. The hybrid function conversion for joystick input is implemented in this module. In addition, the display module shows the wheelchair status such as battery status, obstacle information, driving distance, etc. Especially, this module also provides the serial communication interface to connect with a personal computer or PDA.
2. Ultrasonic sensing module: this module collects seven ultrasonic sensor data.
3. Goal-seeking module: this module implements the fuzzy logic based goal-seeking navigation function
4. Wall-following module: this module realizes the fuzzy logic based wall-following navigation function
5. Obstacle avoidance module: this module implements the fuzzy logic based obstacle avoidance navigation function.
6. Module fusion module: this module realizes the navigation fusions as shown in Fig. 7 and Table 1.

7. Central control module: this module is responsible of coordinating the user joystick command, ultrasonic sensor data and autonomous navigation functions.
8. Velocity justification module: this module implements the fuzzy logic based velocity justification function.
9. Kinematics module: this module converts the steering coefficient and the average velocity as angular velocities of two motors.
10. Closed loop motor control module: this module implements the closed loop angular velocity controller in terms of proportional-integral-differential (PID) control algorithm.

Finally, the power drive module is responsible of driving the wheelchair motors. In addition, based on the proposed architecture, the user may drive the intelligent robotic wheelchair like as a conventional powered wheelchair; however, the goal-seeking, wall-following, obstacle avoidance, and velocity control functions are continuously executing at the backend to perform more comfortable and safe driving.

Based on the proposed distributed computing architecture, a prototype is fabricated in laboratory. Fig. 16 shows the control board. The size of this controller is 10 cm * 15 cm. Finally, a robotic wheelchair is constructed as shown in Fig. 17.

Fig. 16. Photo of the proposed distributed PSoC based controller.

Fig. 17. Photo of an assembled robotic wheelchair.

6. Experiments and verifications

The experiments focuses on the validations of the proposed PSoC based distributed computing architecture. Initially, the PC based simulation environment is constructed to generate correct navigation results, as shown in Fig. 18. These results are used to valid individual navigations functions. The PC based navigation program is developed based on the navigation fusion architecture shown in Fig. 7 and Table 1.
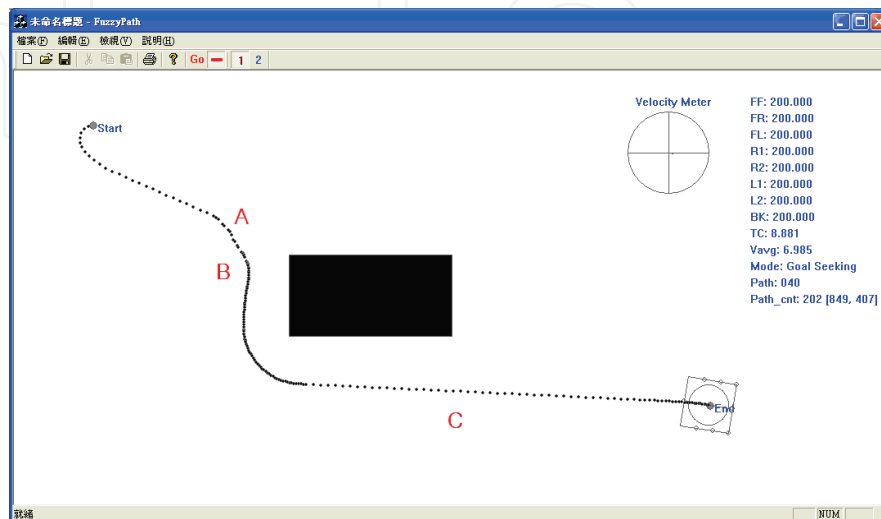

Fig. 18. PC based simulation program for autonomous navigations.

This program is coded using the Microsoft Visual C++ (Kruglinski, 1996). A black rectangle emulates the obstacles. The start and end points are indicated on the screen. The robotic wheelchair executes the goal-seeking mode initially.

When the robot closes to the obstacle, the obstacle avoidance mode takes over (point A). Subsequently, the wall-following mode executes (point B) to follow the left and bottom sides of the obstacle. Finally, the goal-seeking mode executes again (point C) to real the end point. Note that the path history of the robotic wheelchair is represented using the "dot". These dots are recorded in constant time. Therefore, the effects of determining average velocity can be obviously observed from the dots' density on path. Such an effect validates the fuzzy model of the average velocity function. In addition, the robotic wheelchair avoids collisions all the way.

The next step is to validate the correctness of individual PSoC based navigation functions. In this experiment, the PC based simulation program and the PSoC based computing architecture are executing in parallel, and they use identical simulation conditions with the same start and end points. The inference results of the PSoC based computing modules are transmitted to the PC system. Therefore, the screen displays the results of two wheelchairs and paths: blue path is plotted for PC based navigation results; and red path is plotted for PSoC based navigation results.

Results of PSoC based inference are compared with PC based solution. Fig. 19 shows the comparison results of the goal-seeking navigation function. In this figure, the block located at the left-upper corner indicates the operated joystick coordinates. The list box located at the left-lower corner shows the replied steering coefficient of the PSoC controller. The circle located at the right-upper corner represents the velocity meter that contains the direction and value to the wheelchair moving velocity.
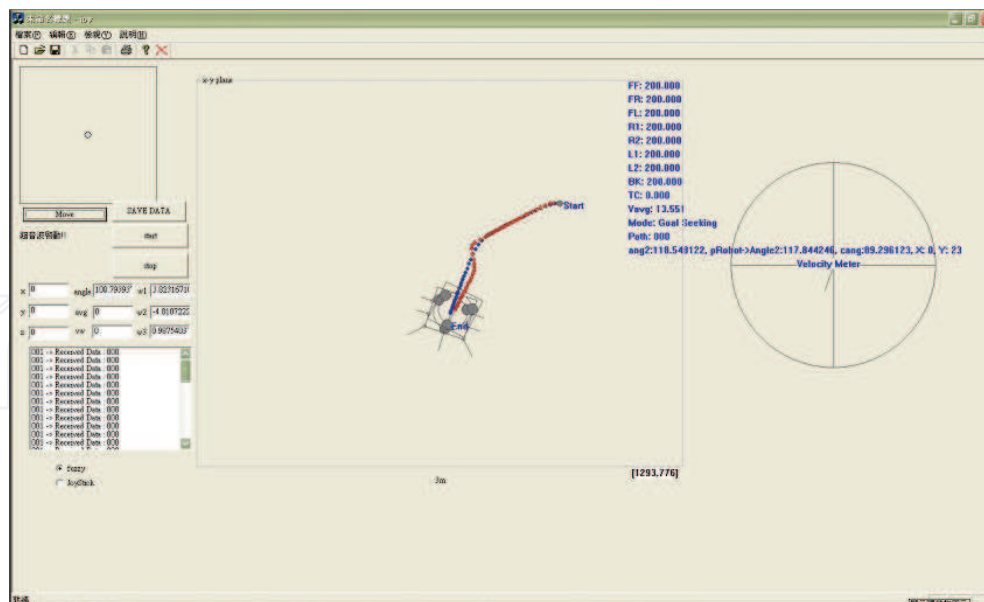
Fig. 19. Comparisons of goal-seeking navigation function.

In addition to the goal-seeking function, the wall-following and obstacle avoidance functions are also validated as shown in Fig. 20 and Fig. 21 respectively. Apparently, the blue and red path can all reach the desired goal; however, the PC based solution is more accurate than the PSoC based solution since the limited digital resolutions of the 8-bits processors. Nevertheless, the PSoC solution is also applicable in real applications.
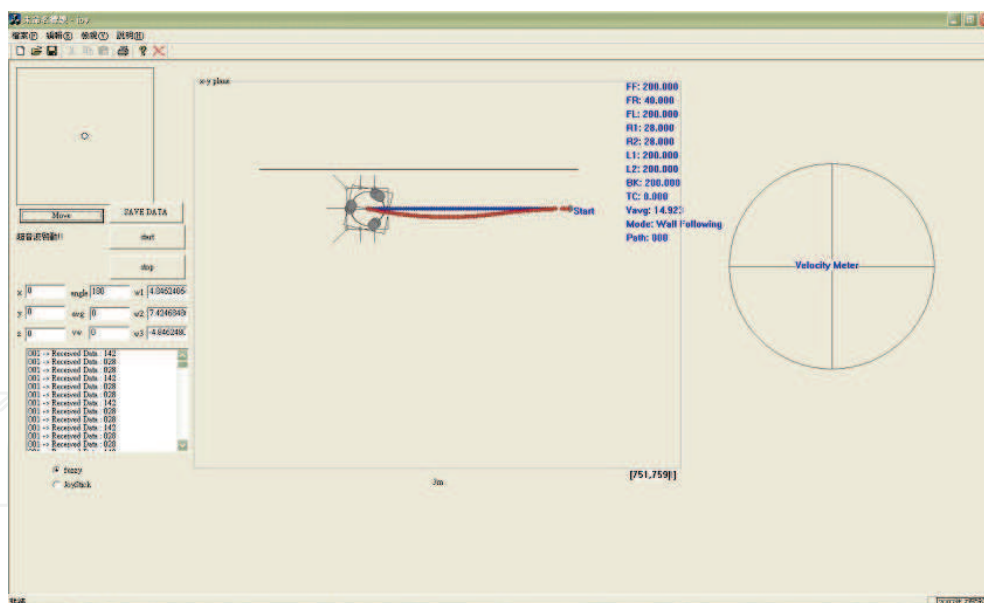

Fig. 20. Comparisons of wall-following navigation function.

In addition to the accuracy of navigation functions, the real-time property is also evaluated. Table 6 shows the computational time of inferring the fuzzy logic based navigation functions. Note that the time is measured in second, and it includes the communication traffic time. The computation time of a navigation function can be finished within 0.14 second. It is reasonable to the wheelchair users because of the driving speed of the wheelchair being not fast for the user who needs the navigation assisted functions.
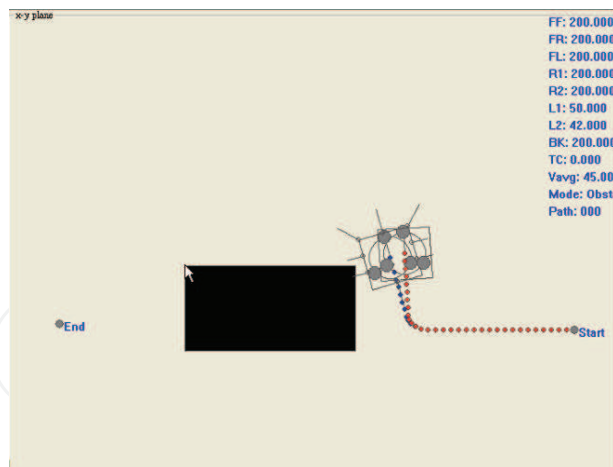
Fig. 21. Comparisons of obstacle avoidance navigation function.

Navigation Function	Computation Time	Error %
Goal-Seeking	0.125 s	8 %
Wall-Following	0.11 s	7 %
Obstacle Avoidance	0.14 s	8 %

Table 6. Computation time evaluations

Finally, a practical test result of plotting the path of the wheelchair on the ground is evaluated in terms of different maximum wheelchair velocities (20%, 40%, 60% and 80%). The blue line indicates the wall of this experiment. Form the results of experiments, the wall-following fuzzy navigation function can work properly. Due to computational time limitations of the chip, large velocity results in a delay in sensor response and larger error.

7. Conclusions and future works

This chapter presents the navigation technologies of mobility assistive robots for disabled patients. The autonomous navigations of conventional robotic wheelchairs are not easy to be implemented because the inconvenience of setting target positions for each segmental path, and unavailable global localization and map of environment. Therefore, a feasible robotic wheelchair is achieved by controlling the wheelchair using the joystick, and the hybrid function converts joystick command as the virtual target for the fuzzy based navigation functions. In this manner, the proposed navigation functions become important assistive function to justify user's command and to perform more safe and comfortable wheelchair driving mechanism. On the other hand, a PSoC based distributed computing architecture is implemented to control the robotic wheelchairs. The proposed architecture is proposed to reduce the costs, size, and power consumptions and to increase the reliability of the popular PC based navigation system. Experiment results validated the PSoC based navigation functions. Based on practical experiments, the computational error is within 8 % and the time elapsed for computation and communication traffic is within 0.14 second. More specially, the proposed architecture is more feasible than the PC based navigation system. In the future, the navigation algorithm will be optimized to increase the computational efficiency and accuracy as well. At the same, in-hospital test will be planned to verify the clinical effectiveness.
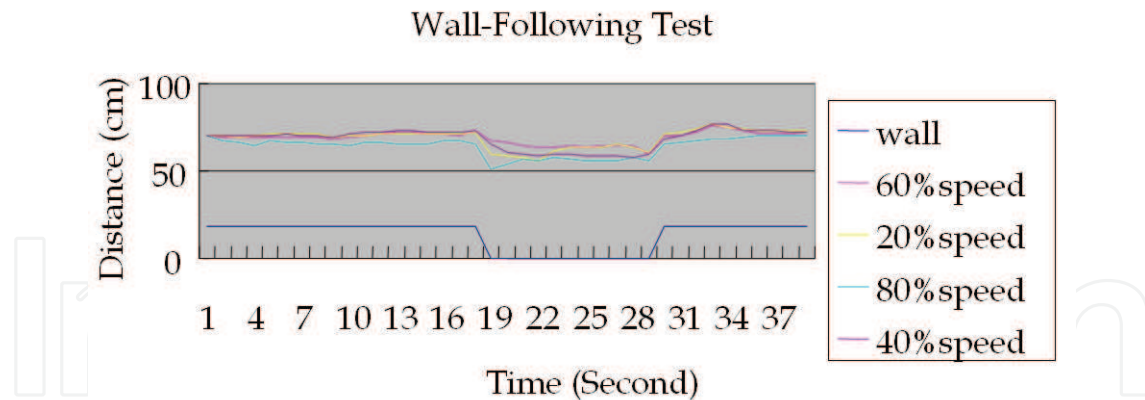
Wall-Following Test

Fig. 21. Comparisons of obstacle avoidance navigation function.

8. References

Asai, H. (2001). Motion Planning of Obstacle Avoidance Based on Guide Style for Autonomous Mobile Robot, *Proceedings of the Seventh International Conference on Virtual Systems and Multimedia*, pp. 694 -700, Berkeley, California, U.S.A.

Bourhis, G.; Horn, O.; Habert, O.; & Pruski, A. (1997). The Autonomous Mobile Robot SENARIO: A sensor Aided Intelligent Navigation System for Powered Wheelchairs. *IEEE Robotics & Automation Magazine,* Vol. 4, 60-70.

Bourhis, G.; Horn, O.; Habert, O.; & Pruski, A. (2001). An Autonomous Vehicle for People with Motor Disabilities. *IEEE Robotics & Automation Magazine,* Vol. 8, 20-28.

Kruglinski, D.J. (1996). *Inside Visual C++ - The Standard Reference for Programming with Microsoft Visual C++ Version 4,* Microsoft, Washington.

Kuo, C.H.; & Chen, H.W. (2006). Human-Oriented Design of Autonomous Navigation Assisted Robotic Wheelchair for Indoor Environments, *Proceedings of IEEE International Conference on Mechatronics*, pp. 230-235, Budapest, Hungary.

Lankenau, A.; & Rofer, T. (2001). A Versatile and Safe Mobility Assistant. *IEEE Robotics & Automation Magazine,* Vol. 8, 29-37.

Luo, R.C.; Chen, T.M.; & Lin, M.H.. (1999). Automatic Guided Intelligent Wheelchair System Using Hierarchical Grey-fuzzy Motion Decision-making Algorithms, *Proceedings of the 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 900-905, Kyongju, Korea.

Mazo, M. (2001). An Integral System for Assisted Mobility. *IEEE Robotics & Automation Magazine,* Vol. 8, 46-56.

Negnevitsky, M. (2005). *Artificial Intelligence – A Guide to Intelligent Systems,* Addison Wesley, New York.

Service Robot Applications

Edited by Yoshihiko Takahashi

The aim of this book is to provide new ideas, original results and practical experiences regarding service robotics. This book provides only a small example of this research activity, but it covers a great deal of what has been done in the field recently. Furthermore, it works as a valuable resource for researchers interested in this field.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Chung-Hsien Kuo (2008). Mobility Assistive Robots for Disabled Patients, Service Robot Applications, Yoshihiko Takahashi (Ed.), ISBN: 978-953-7619-00-8, InTech, Available from: http://www.intechopen.com/books/service_robot_applications/mobility_assistive_robots_for_disabled_patients

INTECH
open science | open minds