Behavior-based Perceptual Navigation for Semi-autonomous Wheelchair Operations

H. Uchiyama and W. D. Potter∗

{hajime, potter}@uga.edu

Artificial Intelligence Center
GSRC 111, Brooks Drive
University of Georgia
Athens, GA 30602-7415
USA

Abstract

This paper describes an overview of our semi-autonomous (SA) wheelchair prototype and emphasizes the design of the perceptual navigation system. The goal of our SA wheelchair project is to increase independent mobility of individuals who are wheelchair-bound and severely vision impaired. The scope of the project focuses on developing the perceptual navigation system with certain autonomy, but not on controlling the movement of the wheelchair in order to enable the user to maintain the maximum control over the wheelchair movement. We present an overview of a customized behavior-based architecture: Behavior Cell and Behavior Network components, which are designed to add an extended input/output feature.

1 Introduction

Studies have shown that individuals with disabilities, regardless of their age, benefit substantially from access to independent mobility, including electrically powered wheelchairs (Douglass and Ryan, 1987; Paulsson and Christoffersen, 1989; Verburg et al., 1991). Independent mobility increases vocational or educational opportunities, reduces dependence on caregivers, and promotes a feeling of self-reliance and self-esteem. Unfortunately, some individuals are unable to independently operate a powered wheelchair due to their impairments of motor, sensory, perceptual, or a combination of these.

In order to assist those individuals, a number of studies have been conducted in the field of assistive technology which combine robotics and artificial intelligence to develop robotized wheelchairs; some try to achieve autonomous control over the wheelchairs, and others aim for semi-autonomous control. Most of these robotized wheelchairs are equipped with a computer and a set of sensors, such as cameras, infrared sensors, ultrasonic sensors, and laser rangers. This assortment of equipment is used to address a number of specific problems such as: obstacle avoidance, local environment mapping, and route navigation.

The CALL Centre of the University of Edinburgh has developed a wheelchair intended to be used by children who do not have the physical, perceptual or cognitive abilities to control an ordinary powered mobility aid (Odor and Watson, 1994)1. The Smart Wheelchair exhibits collision detection followed by a maneuvering action, following the line tracks laid along the floor, and communication between the user via a speech synthesiser or other feedback system. The TAO series (Applied AI Systems Inc.) is mainly designed for research and development purposes. TAO-7 is equipped with a voice recognition/synthesis interface and performs free-space detection in a crowd and landmark-based navigation in addition to the common tasks such as obstacle avoidance (Gomi and Griffith, 1998). The Tin Man project at the Kiss Institute is aimed at the development for a low-cost robotic wheelchair to aid people with impaired mobility (Miller and Slack, 1995). Tin Man II exhibits manual navigation with obstacle avoidance override, autonomous control, and manual mode. The Wheelesley project at MIT intends to be used by users unable to manipulate the standard motorized wheelchair and aims to establish the system for both indoor and outdoor navigation. Wheelesley was developed based on the Tin Man model and interacts with the seated user via a graphical user interface (Yanco, 1998). The NavChair system (University of Michigan) , one of the most successful systems in the ‘90s, intends to provide mobility support to users who are unable to drive standard motorized wheelchairs by autonomously selecting three different modes (tasks): obstacle avoidance, door passage, and wall following (Levine et al., 1999).

∗Contact author: 1-706-542-0361 (office), 1-706-542-2966 (FAX)
1Information is available at http://callcentre.education.ed.ac.uk/
Despite these efforts, very few robotized wheelchairs are currently observed in the market, and none are intended to be used outside of a research lab or a training facility. The above studies are mainly issuing autonomous control system. With autonomous control, the system probes the environment, detects an obstacle, plans a navigation route, makes decisions, and fully controls the mobility of the wheelchair, which leaves the user totally dependent upon the equipment.

While some users might be comfortable with an autonomous wheelchair transportation system, others want to be more involved with the process. It is essential for them to feel as if, or actually, they are in control, while being responsible for both the decision-making and motion of everyday transportation activities rather than being a passenger.

1.1 Semi-autonomous Wheelchairs

The purpose of developing a semi-autonomous (SA) wheelchair system, while it comprises hardware equipment similar to autonomous wheelchairs, focuses on providing just as much assistance as the user really needs, in order to enhance the level of autonomy of the user. Generally, the type of assistance for a user varies from one to another; therefore, the specification of an SA wheelchair system needs to be determined based on the user’s physical conditions.

A number of research projects have been actively conducted to provide various types of semi-autonomous wheelchairs. Borgolte et al. (1998) provided omnidirectional maneuverability to allow an intuitive semi-autonomous control system to assist people with severe or multiple handicap. Argyros et al. (2002) presented a semi-autonomous navigation system comprising a panoramic vision system to support the users who have limited motor control of the upper extremities. The Mobile Internet Connected Assistant (MICA) project (Rönnbäck et al., 2006) designed a system in which the user either locally or remotely controls the chair by means of head-held sensors. Despite using a slightly different platform, a power-assisted manual wheelchair\(^2\), Simpson et al. (2005) demonstrated the first prototype which attempts to provide users who have visual impairments with some basic navigation tasks such as collision avoidance.

However, none of these systems provide or enhance perceptual capabilities for the seated user who has severe visual impairment. The goal of our SA wheelchair project is to increase independent mobility of individuals who are wheelchair-bound and severely vision impaired by providing a perceptual navigation system. The scope of the project focuses on developing the perceptual navigation system with certain autonomy, but not on controlling the movement of the wheelchair, in order to enable the user to maintain the maximum control over the wheelchair movement.

2 Design

2.1 Behavior Cell Approach

Since the SA wheelchair system needs to be individualized to the user, our design principle shall be based upon modularity and flexibility. Further, due to the nature of the navigational tasks required during wheelchair operations, which must be resolved quickly and handled concurrently, the behavior-based control (BBC) architecture (Arkin, 1998; Brooks, 1991; Matarić, 1992) is suitable for the base architecture of our design. Utilizing a customized BBC architecture, we define behaviors which emerge from an agent-environment interaction based on a number of loosely tied processes that run asynchronously and respond to problems in a parallel manner.

In our project, behaviors are realized to accomplish their tasks through user-machine cooperation. Two cooperating behaviors, the Perceptual Behaviors and Navigational Behaviors, acquire the state of the environment through sensors, interpret or responds to the state, and pass the navigational signals to the user. Finally the user manipulates the wheelchair in combination with his/her own perception and judgement (Manipulative Behaviors).

2.1.1 Behavior Cell

We present a unit of the behavioral structure, a Behavior Cell, which is based upon the BBC architecture with the extended input/output feature. A Behavior Cell consists of an input/output (I/O) component, a behavioral function component, and an internal storage component (Figure 1). It structurally resembles an artificial neuron; however, it has a logical gate in addition to widely extended functions such that the innervation link between cells can run by both Boolean and numeric means.

The I/O component consists of a subset of the I/O ports characterized as follows: Port-EB, excitatory inputs; Port-IB, inhibitory inputs; Port-DI, sensory/behavioral inputs; Port-RS, a reset signal; Port-IS, an innervation signal;\(^2\)

\(^2\)Based on a manual wheelchair, the rear wheel hubs are replaced with motorized ones that magnify and/or adjust the propulsive force provided by the user.
Port-AB, an activation output; Port-EO, an effect output; and Port-AO, actuator outputs. The excitatory and inhibitory inputs are linked to the corresponding behaviors’ activation output ports. When any activation/inhibition conditions are met, the behavior is activated/deactivated. Our architecture allows both Port-EB and Port-IB to specify activation (inhibition) conditions by using logical expressions.

Port-DI takes various types of data inputs from sensors or other behaviors (effect outputs). When Port-IS receives an innervation signal from outside or from Port-RS, the behavior checks or sends its inputs and outputs. If Port-RS receives a reset signal from the system, the behavior will clear all dynamic contents of the storage component and activate Port-IS. Port-AB contains an activation value (binary) that is linked to the value of Port-EB. Port-EO contains an effect value that is derived from the behavioral function. If the behavior is connected to its effector(s), Port-AO sends Action Outputs to them.

The behavioral function component provides a flexible activation/computation functionality, such as algebraic sum, sigmoid, Gaussian, and logical expressions, as well as a simple by-pass function (e.g. a direct link between inputs and outputs). More complicated functionalities, such as fuzzy logic inference operators or artificial neural networks, can also be implemented.

The storage component provides a storing capability of the current state onto its dynamic data, which enables the behavior to achieve goals that contain temporal sequences. It may also contain internal static data which all instantiated behaviors can share and refer to, as well as individual constant data that a behavior utilizes as permanent reference information, such as threshold values or look-up tables.

The activation/computation process performed by a Behavior Cell is as follows:

1. When Initialization/Reset input (Port-RS) is activated, it refreshes the internal dynamic memory and innerves Innervation Input (Port-IS).

2. When Innervation Input is innerved, check the value of Effect Inputs (Port-EB). If true, set Activation Output (Port-AB) value to 1 (true) and go to the next step, otherwise return.

3. Check the value of Inhibitory Inputs (Port-IB) to see whether the behavior is inhibited. If false, go to the next step, otherwise set Activation Output (Port-AB) to 0 (false) and return.

4. **In case of using Port-EO:** Using the information from Sensors/Behavior Inputs (Port-DI), derive the return value from the behavioral function and write this value to Effect Output (Port-EO) and return. Store the necessary data in the internal memory if so designed.

5. **In case of using Port-AO:** Similar to (4), derive the return action commands from the behavioral function and send the commands to the effectors via Action Outputs (Port-AO) and return. Store the necessary data in the internal memory if so designed.

### 2.1.2 Behavior Network

Similar to other Behavior-Based architectures (for instance, Nicolescu and Matarić, 2002), our approach also enables behaviors to consist of other behaviors (Behavior Network). In a Behavior Network, behaviors communicate with each other through their port-to-port links, and precondition dependence characterizes the links; thus, the activation of a behavior is dependent on its pre-conditional links.

An individual Behavior Cell can connect to multiple Behavior Cells, and similarly, multiple Behavior Cells can be linked to a single Behavior Cell. This multiple-connectivity allows a Behavior Cell to be a member of multiple Behavior Networks, which makes component-like behaviors, such as interface to the sensors, reusable. Containing multiple task-oriented/reactive behaviors (functional behaviors) enables a Behavior Network to accomplish various
tasks, such as command arbitration, learning, and planning, while asynchronously performing tasks within the distributed architecture.

A Behavior Network shall also behave as structurally equivalent as a Behavior Cell when observed from outside. In order to do so, each Behavior Network must contain a specific type of Behavior Cell (I/O cell) which accomplishes the tasks related to input/output communication and activation sequence inside of the Behavior Network. Figure 2 depicts a generic Behavior Network that consists of I/O cells and functional behaviors.

The I/O cells are categorized as Boolean I/O cells that exchange Boolean signals and activation I/O cells that control sequential activation in the Behavior Network. Figure 3a illustrates a generic Boolean I/O cell that consists of Port-EB (excitatory inputs), Port-RS (reset signal), Port-IS (innervation signal), Port-AB (activation output), and a Boolean algebraic function. The Boolean algebraic function can employ various symbolic logic expressions, and the result from the function is directly connected to the activation output (Port-AB).

Activation I/O cells are responsible for the sequence of innervation/reset functionalities in the Behavior Network. An activation I/O cell consists of an excitatory input, an inhibitory input, an innervation input, an activation output, action outputs, and a storage component that contains a predefined activation sequence of the behaviors (Figure 3b). The activation I/O cell innervates the reset/innervation ports of the behaviors that belong to the Behavior Network according to the sequence stored on the storage component.

The functional behaviors deal with data and/or actuator communication. Connections between functional behaviors consist of excitatory links (between Port-AB and Port-EB), inhibitory links (between Port-AB and Port-IB), (sensory and/or behavioral) data links (between Port-EO and Port-DI), and actuator links (between Port-AO and effectors). A functional behavior in a Behavior Network can also be another Behavior Network.

However, the activation process of a Behavior Network differs from the one of a Behavior Cell. An innervation cell is first innervated and it innervates the Excitatory Inputs cell, Inhibitory Inputs cell, and Activation Output cell in this order. If the value of Port-AO (activation output) of an Activation Output cell is true, it will innervate the functional behaviors in a predefined order otherwise the whole process will return; thus the Behavior Network will be deactivated.

2.2 System Components

In this paper, we briefly overview the system components of our SA wheelchair prototype. The major add-on hardware to the base wheelchair (Invacare Nutron® R32) comprises stationary ranging modules, motorized vision modules, motion sensing modules, and the tactile feedback module.

The stationary ranging module is a collection of ultrasonic sensors (Devantech SRF-08) which are attached to the wheelchair and constantly measure distances to any objects that are within the scope of the sensors. Six sonars...
are equipped to the front side, and two sonars are placed to the rear side, as illustrated in Figure 4a. We designed a customized motorized vision module (Figure 4b) which consists of a webcam (Logitech QuickCam® Pro 4000), laser line generator, and tilt/pan servo motors. It is capable of tilting (0–135°) and panning (0–360°) and acquires depth data as well as scenery images. The motion sensing module consists of a gyroscope and accelerometers and continuously tracks the acceleration and angular velocity of the wheelchair. The above sensor system is controlled by a micro-controller (Rowely Associates Ltd. CrossFire® LPC-2138 evaluation kit) which is connected to the host computer via a USB.

The tactile feedback module is designed to convey vibrotactile signals to assist users to manipulate the SA wheelchair. We designed a Vibrotactile Glove which consists of an array of vibrotactors inside the glove on the back side of the hand (Figure 5). Each vibrotactor is directly mapped to correspond to the ultrasonic sensors as shown in Figure 4a. A vibrotactor generates a vibrotactile stimulus consisting of a rapid vibration lateral to the skin surface, and its characteristic is determined by the duration of stimulus and the inter-stimuli interval. A sequential stimulus consists of a sequence of multiple vibrotactile stimuli generated by different vibrotactors. We also designed a motor-array controller which controls the vibrotactors based on commands sent from the host computer via a RS-232C.

3 Behavior Implementation

The perceptual behaviors of interest in this paper are Obstacle Notification, Free-space Finding, and Doorway Navigation, each of which is a Behavior Network consisting of a set of other Behavior Cells or Behavior Networks. Navigational behavior is represented as the Navigation Command Manager, and the Sensor Command Manager manages the commands for the sensor-related effectors (servo motors and the laser line generator). Figure 6 illustrates the schematic diagram of the perceptual and navigational behaviors accompanied with the sensor-effector systems. The thin arrows represent data links, and the thick arrows represent actuator links. Other port-to-port links are not shown. In this paper, we briefly describe the implementation of Obstacle Notification, Free-space Finding, and Doorway Navigation behaviors.
3.1 Obstacle Notification

The Obstacle Notification behavior is one of the most reactive behaviors, which takes sensor readings from the ultrasonic sensors and directly maps into vibrotactile representations. It consists of two subordinate behaviors: Sonar Reader and Obstacle Detection. The Sonar Reader behavior is a component-like behavior which reads sensor data from the micro-controller. The Obstacle Detection behavior interprets the sensor data into a vibrotactile representation which expresses the distance of an obstacle in each sensor.

3.2 Free-space Finding

The Free-space Finding is also a reactive behavior relating to the sensor readings but behaves oppositely to the Obstacle Notification behavior. Instead of notifying obstacles, it reports a band of orientation in which no immediate obstacles are found. This behavior is particularly effective to find a way out when the SA wheelchair is surrounded by a crowd of obstacles. It consists of Sonar Reader, Way finder, and Range Finder behaviors. It primarily relies on the sonar readings, however, if the sonar readings do not provide enough information, it invokes the Range Finder behavior to acquire more accurate depth information.

3.3 Doorway Navigation

The task of passing through a doorway, in general, comprises several subtasks addressing dynamic determination of a maneuvering trajectory and decision making which relies on high level cognitive processes. Some previous studies demonstrate a creation of local map in order to generate a maneuvering trajectory for an autonomous vehicle (Patel et al., 2002; Surmann et al., 2003). Such maneuvering trajectories may shape a complicated curvature which may not be appropriate to represent to the user via a sequence of tactile signals. We propose a simple guidance approach that utilizes the Pivotal Point/Zone, an intermediate goal point (area) from which the wheelchair can straightforwardly move toward the doorway.

The Doorway Navigation behavior comprises the following behaviors: Doorframe Detection, Range Finder, Object Tracker, and Path Planner. The Doorframe Detection behavior searches a doorframe from the image stream, and if it finds a candidate, its orientation data is passed to the Range Finder behavior to confirm the validation of the candidate. Once the doorframe is confirmed, the Object Tracker behavior traces the door in order to fixate the camera to the doorframe. In the mean time, the Path Planner behavior is invoked to undertake a repetitive process of localizing the wheelchair, sending a signal indicating the desired orientation to the user, and adjusting the path plan, until the wheelchair reaches the sub-goal (pivotal point). At the pivotal point, the user is guided to swivel until the wheelchair straightforwardly faces to the doorway and then to move forward.

4 Current Status

Currently we are in the process of implementing the behaviors and are close to the completion. Experimental field tests of the SA wheelchair operation are underway.
5 Conclusion

In this paper, we proposed a customized BBC architecture to design the perceptual behaviors for the SA wheelchair operations. Although the experimental field tests are still underway, the Behavior Cell and Behavior Network components provides flexibility, extensibility, and polymorphism in design which enables more intuitive implementation of a planning oriented task for our SA wheelchair project compared to a standard BBC architecture.

References


