

Research report

Tamara Blain

1 Introduction

The goal of this project is to extend an approach to shared control of a robotic wheelchair, which is centered around a maps-based interface [1] and the simultaneous localization and mapping (SLAM) algorithm [2]. Continuing research into non-invasive BCI's have enabled people with severe neuromuscular disabilities, more control over important aspects of their lives [3]. One of the most important, is control over their mobility. Successful strategies have been employed, allowing users to steer wheelchairs with EEG signals [4], [5], [1]. But EEG signals are not very reliable, as they are inherently noisy. In addition, depending on the nature of the disability, varying degrees of difficulty may be encountered in complete control of a mobile vehicle. Finally, the complexity of everyday environments, makes it a challenge to design systems which allow for mental control but are still robust enough to be able to cope with such complexity [3]. Hence, adaptive shared control strategies have been employed in the control of smart wheelchairs [6], [7]. These strategies provide the most flexibility in allowing a wider range of mobility-limited users with varying degrees of cognitive capabilities, to control their wheelchairs using brain signals. Shared control is also more suitable for the restoration of a wheelchair user's sense of independence [7]. This project extends the work of [1], and presents additional options to present to the user, which help facilitate their mobility. An alternate implementation of the SLAM algorithm using a particle filter is also part of this approach.

2 Background

2.1 Smart Wheelchairs

In [6], robotic wheelchairs are classified by the way control is allocated between themselves and the user. The ones most similar to autonomous robots usually require a complete map of the environment through which they navigate, or modification to the environment such as tape tracks [6]. They plan and execute paths to destinations chosen by the user in known controlled environments [6]. This class of smart wheelchairs is best suited for users who experience difficulty planning and/or executing paths [6]. An alternative class consists of wheelchairs whose assistance is limited to obstacle detection and avoidance, and require most of the planning and navigation be left to the user [6]. They require more continuous effort on the part of the user [6]. A third class allows for shared control between the user and the wheelchair. The newest and most promising class is that of the adaptive shared control wheelchairs [6], [7]. They offer multiple context-based operating modes, and seem the most flexible and practical.

2.1.1 EEG

Electroencephalography (EEG) uses electrodes which sit on the scalp, to measure voltage fluctuations in the brain. The number of sensors used can range from 25 to 128 sensors. Typically, a small area of the scalp is lightly abraded right before the application of a sensor. This is done to reduce the impedance caused by dead skin cells. A conductive gel is applied with each sensor, and each sensor is connected to a wire which feeds into an amplifier. Scalp EEG is widely used in diagnosing many medical conditions such as epilepsy, strokes, and sleep disorders [8]. But translating those signals into meaningful information is very difficult. One of the largest obstacles is the noise. EEG has inherently low spatial resolution, so the signals picked up from EEG sensors reflect the activity of millions of neurons [8]. In addition to brain activity, electrical signals from the scalp can also come from sources such as eye blinks or tongue movement [8]. Electrical impulses from the heart can actually produce scalp potentials greater than EEG amplitudes [8], thus making recorded signals even more difficult to discriminate.

2.2 BCI

A brain computer interface (BCI) is a communication channel between the brain and a computer. It records changes in electrical activity in the brain, and translates them into control signals. Acquired signals are first amplified, then digitized for analysis. Unwanted artifacts, such as those produced by a heartbeat, and eye blinks, are filtered first amplified target signal feature is extracted [9]. Built-in classifiers characterize brain state by correlating specific mental activities with the target features [9]. The interpreted brain state is commuted into control signals (a single bit in this case), and sent to the device being operated. BCI's can be either invasive, such as single-unit recording or electrocorticography (EcoG), or non-invasive, such as EEG or MEG. Invasive techniques provide the most information per channel, but carry the greatest risk, as they require surgery to provide openings in the skull. Most BCI systems are non-invasive, and use EEG signals recorded from the scalp [10]. But typically, scalp-recorded EEG-based BCIs tend to be synchronous; they tend to be time-locked to cues, resulting in very low bandwidth [11].

2.3 Extended Kalman Filter (EKF)

A Kalman filter is a technique used for state estimation in probabilistic temporal models. It assumes a linear gaussian transition model: a linear gaussian distribution is used to represent the transition model, and gaussian noise is assumed [12]. Extended Kalman filters were designed to deal with non-linearities in the system being modeled [12]. The extended version works by modeling the system as locally linear near the mean of the current state distribution [12]. Both versions are limited by their ability to only model state distributions as a single multivariate Gaussian, while many real world applications require the flexibility of being able to model arbitrary distributions [12].

2.4 Particle Filtering

Particle filtering is a family of sampling algorithms which use the samples themselves to approximate state distributions in bayesian models [12]. It works because samples are weighted according to the observations, and those samples in the space where the posterior is low (those with low weights), are discarded [12]. In this way, the samples stay fairly close to reality. They are advantageous in that they are able to model hybrid distributions [12].

2.5 Simultaneous Localization and Mapping (SLAM)

SLAM is an algorithm for autonomous mobile agents involving the generation of a probabilistic map of the agent's environment, finding its pose within that map, and the use of information in the map for the purpose of navigation [13]. It assumes a known kinematic model, a linear discrete time model, and sensors which can take measurements of locations between any landmark and the autonomous agent itself [2]. In the original solution to SLAM [2], the use of extended Kalman filters (EKF) for solving the SLAM problem is proposed.

3 Related Work

The MAPs BCI interface is a graphical interface which allows a user shared control of an autonomous mobile agent [1]. The interface is a hierarchy of operating levels, each of which requires different degrees of control from the user. On one level, the user is able to navigate freely using only brain signals. A control strategy which includes obstacle avoidance, is employed while the user navigates the wheelchair freely [1]. Internal representations of the environment (maps) [14] are generated by the agent using the simultaneous location and mapping algorithm (SLAM), while it is controlled solely through the BCI [1]. A grid map of the environment is built using the EKF-based SLAM algorithm [1]. The grid map is used to build a probabilistic map, which is then used to build an evidence grid [1]. This grid is then stored in its database as a map.

Another level presents the user the option to select from the stored maps. Options are presented to the user as a grid of cells which are flashed sequentially [1]. A selection is made when a specific brain event is detected by the BCI. Once a map is selected, the user can generate trajectories to navigate through it by selecting the initial and final positions of the mobile agent [1].

4 My Approach

This project proposes improvements to the MAPs-BCI interface. The first modification is relatively simple and is aimed at reducing the time it takes the user to get to a known landmark. In the MAPs-BCI interface, the user can generate trajectories to navigate through a map by selecting the initial and final positions of the mobile agent [1]. The modification involves appending an acquired image of a landmark, to its location stored in memory. This could be done during the mapping process. The idea is to allow users to choose from among images of known landmarks and further rely on the wheelchair's already present localizing ability. This would further reduce the load on the user, and reduce the time it took to get to a landmark, by reducing the number of steps from three (choose map, initial position, final position) to one. It could be implemented as a separate level offered to the user or a replacement to map selection altogether.

The second contribution is an application of the work addressed in [13]. The MAPs-BCI interface uses the EKF-based SLAM algorithm in order to orient itself and create a stochastic map of the environment when it is in an unknown environment (an environment with no corresponding map in its database) [1]. This modification consists of replacing the SLAM algorithm with a version which is more accurate than the original - FastSLAM [13]. FastSLAM relies on motion estimation and data

from the most recent sensor measurement to estimate an agents pose [13]. It also uses the fact that landmark estimates are conditionally independent of each other given the sensor measurement, to factor the posterior into a form which allows each particle to be associated with its own path.

5 Work to date

Thus far, I have built a simulation in python, of an autonomous agent equipped with sonar and an encoder, which constructs a deterministic evidence grid of a room. In simulation, the agent enters the room and immediately turns left. It then proceeds forward for one step, stops and takes a reading to the left of it. It records whether or not a cell is occupied, then takes a reading directly ahead. If the closest obstacle ahead is within a threshold, it turns right and proceeds forward. Else, it simply proceeds forward. This process is repeated as the agent makes its way around the room. The step size is the size of the agent itself (a few pixels, in this case).

In a separate routine, the agent, now outfitted with GPS, is placed randomly within the room. The random location is first tested for validity, i.e. not inside an obstacle. A naive particle filter explained presently, is used to determine its position within the room. Initially, its position is assumed to be anywhere in the room, with all positions equally likely. Thus, the initial set of particles are sampled from a uniform distribution. Each sample is first tested for validity. Then, the standard deviation (SD) of each sample is compared to the standard deviation of the GPS distribution. If the ratio of particle SD to GPS SD is more than three, the particle is discarded. The remaining particles are re-sampled. GPS sensor readings are modeled as a gaussian centered at the actual location of the agent.

6 Future Work

GPS readings are only useful for outdoor situations. The GPS will be replaced by a range sensor such as an IR range finder. A particle filter such as that used in the FastSLAM [13] will be implemented, for increased accuracy in localization. Also, as the MAPs algorithm requires a probabilistic map, a stochastic mapping technique will be integrated into this approach. Finally, the association of images with landmarks will be introduced.

References

- [1] F.A. Cheen, R. Carelli, W.C. Celeste, T.F. Bastos, and F. di Sciascio, "Maps managing interface design for a mobile robot navigation governed by a BCI," *Journal of Physics: Conference Series*, 2007, p. 012088.
- [2] M. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, 2001, pp. 229–241.
- [3] G. Vanacker, J.R. Millan, E. Lew, P.W. Ferrez, F.G. Moles, J. Philips, H. Van Brussel, and M.

- Nuttin, "Context-based filtering for assisted brain-actuated wheelchair driving," *Computational Intelligence and Neuroscience*, vol. 2007, 2007, p. 25130.
- [4] K. Tanaka, K. Matsunaga, and H. Wang, "Electroencephalogram-based control of an electric wheelchair," *IEEE Transactions on Robotics*, vol. 21, 2005, pp. 762-766.
- [5] R. Leeb, D. Friedman, G.R. Muller-Putz, R. Scherer, M. Slater, and G. Pfurtscheller, "Self-Paced(Asynchronous) BCI Control of a Wheelchair in Virtual Environments: A Case Study with a Tetraplegic," *Computational Intelligence and Neuroscience*, vol. 2007, 2007, p. 79642.
- [6] S.P. Levine, D.A. Bell, L.A. Jaros, R.C. Simpson, Y. Koren, and J. Borenstein, "The navchair assistive wheelchair navigation system," *IEEE Transactions on Rehabilitation Engineering*, vol. 7, 1999, pp. 443-451.
- [7] J. Philips, J.R. Millán, G. Vanacker, E. Lew, F. Galán, P.W. Ferrez, H. Van Brussel, and M. Nuttin, "Adaptive shared control of a brain-actuated simulated wheelchair," *Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics*, 2007, pp. 408-414.
- [8] P.L. Nunez and R. Srinivasan, *Electric Fields of the Brain: The Neurophysics of EEG*, Oxford University Press, USA, 2005.
- [9] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for eeg-based brain-computer interfaces," *Journal of Neural Engineering*, vol. 4, 2007, pp. R1-R13.
- [10] C. Guger, A. Schlogl, D. Walterspacher, and G. Pfurtscheller, "Design of an EEG-based Brain-Computer Interface (BCI) from Standard Components running in Real-time under Windows" *Biomedical Engineering*, vol. 44, 1999, pp. 12-16.
- [11] J.R. Millán, F. Renkens, J. Mourino, and W. Gerstner, "Non-invasive brain-actuated control of a mobile robot," *Proceedings of the 18th international joint conference on Artificial intelligence*, 2003.
- [12] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, 2002.
- [13] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," *International Joint Conference on Artificial Intelligence*, 2003, pp. 1151-1156.
- [14] Zunino, G.; Christensen, H.I., "Simultaneous localization and mapping in domestic environments," *Multisensor Fusion and Integration for Intelligent Systems, 2001. MFI 2001. International Conference on*, vol., no., pp. 67-72, 2001