
A self-paced online BCI for mobile robot control

Tao Geng*, John Q. Gan and Huosheng Hu

Department of Computing & Electronic Systems,
University of Essex,
Colchester, CO4 3SQ, UK
E-mail: runbot05@gmail.com
E-mail: jqgan@essex.ac.uk
E-mail: hhu@essex.ac.uk
*Corresponding author

Abstract: This paper presents the design and online experiments of a self-paced online brain-computer interface (BCI) for controlling a simulated robot in an indoor environment. Three one-vs-rest linear discriminant analysis (LDA) classifiers are combined to control the switching between automatic control (AC) and subject control (SC) modes. The hierarchical structure of the controller allows the most reliable class (mental task) in a specific subject to play a dominant role in the robot control. A group of simple rules triggered by local sensor signals are designed for safety and obstacle avoidance in the AC mode. Due to the intuitive nature of the controller and the small number of AC rules, a subject has much flexibility and full control of the robot. Online experiments have shown that subjects successfully control the robot to circumnavigate obstacles and reach some specified targets in separate rooms by motor imagery of their hands and feet.

Keywords: brain-computer interface; BCI; brain actuated robot control; electroencephalography; EEG.

Reference to this paper should be made as follows: Geng, T., Gan, J.Q. and Hu, H. (2010) 'A self-paced online BCI for mobile robot control', *Int. J. Advanced Mechatronic Systems*, Vol. 2, Nos. 1/2, pp.28–35.

Biographical notes: Tao Geng received his PhD from Shanghai Jiaotong University, China, in 2003. His research interests include animal motor control and legged robots.

John Q. Gan received his PhD from Southeast University, China, in 1991. He is a Reader in Computer Science at the University of Essex, UK. His research interests are in intelligent systems and robotics, neurofuzzy systems, learning theory and algorithms, pattern recognition, signal and image processing, information fusion, brain-computer interfaces, system modelling and control, and data mining. He has co-authored a book and published over 150 research papers. He is an Associate Editor for *IEEE Transactions on Systems, Man, and Cybernetics – Part B* and in the editorial board of other journals.

Huosheng Hu is a Professor in School of Computer Science and Electronic Engineering at the University of Essex, UK, leading the Human-Centred Robotics Group. He has published over 300 papers in journals, books and conferences and received a number of best paper awards. He is one of the Founding Members of IEEE Robotics and Automation Society Technical Committee on Networked Robots, a Fellow of IET and a Senior Member of IEEE and ACM.

1 Introduction

Traditional electroencephalography (EEG) based brain-computer interfaces (BCIs) are designed for synchronous operations where a user can only control a robot/wheelchair in time periods predefined and indicated by the system. Although such a system-driven strategy has greatly facilitated the classification of user's intentions in control, a synchronous BCI does not provide a natural mode for human-machine interaction (HMI) in real-world applications and can cause frustration to users. In contrast, self-paced EEG-based BCIs send commands to the controlled devices whenever the user has such an intention.

When there is no control intention, the system outputs remain neutral or unchanged but still available to operate (Borisoff et al., 2006; Sato et al., 2008; Oya et al., 2008; Halder and Sarkar, 2007; Wu et al., 2006).

Since EEG has very low data rates and uncertain patterns, various control strategies have been developed recently. Tanaka et al. (2005) used two classes for turn-left or turn-right, which is insufficient for wheelchair control. Rebsamen et al. (2007) adopted P300 EEG signals to control a wheelchair to reach a target following a predefined path. The big drawback is that the spatial resolution of control is very low. Currently, there are very few robots/wheelchairs that are controlled by self-paced BCIs.

Millan's work is the most successful example. In his asynchronous design (Millan et al., 2004), three-class mental states of a subject are associated with high-level commands only. A finite-state automaton involving more than 30 rules integrates high-level commands and sensor signals to generate real-time commands for controlling the robot. Two subjects in the experiments successfully controlled a small robot (5.7 cm diameter) to visit a few randomly selected rooms in an office-like environment (80 cm \times 60 cm) (Millan et al., 2004).

This paper presents the design and real-time experiments of a self-paced online BCI for controlling a simulated robot in an indoor environment. This design employs three one-vs-rest linear discriminant analysis (LDA) classifiers to spontaneously classify three motor imagery tasks of a subject, i.e., left hand, right hand and feet, which directly and intuitively control three movements of the robot, i.e., turning left, turning right and moving forward/stop. A simple strategy is designed to generate control commands, which is based on outputs from three LDA classifiers. The controller has a hierarchical structure in which the most reliable class (motor imagery) in the subject is given a higher priority. A group of simple rules triggered by local sensor signals takes effect for safety and obstacle avoidance when the robot is approaching an obstacle or a corner, which are the most ubiquitous objects in an indoor environment. As the number of these rules is very small and they are only active during some periods, the subject still has much flexibility and full control of the robot.

The rest of the paper is organised as follows. Section 2 describes how data analysis is carried out offline. Online self-paced BCI experiments are designed in Section 3, including the mechanism for the switching control of AC/SC modes, control rules in the AC mode and turning control in the SC mode. In Section 4, experimental results are presented to demonstrate the feasibility and performance of the proposed control strategy. Finally, a brief conclusion and future work are given in Section 5.

2 Offline data analysis

In order to collect data to train the classifiers used in the self-paced online BCI, a synchronous experiment is carried out first, which consists of eight runs with 30 trials each. In each trial, an arrow pointing to left, right or down was displayed from $t = 3$ s. Subjects were instructed to imagine left hand, right hand or feet movements until $t = 8$ s.

Figure 1 shows five-channel bipolar electrodes placed based on the specification of international 10-10 systems. The EEG data was sampled and recorded via a 16-channel EEG amplifier from g.tec at 250 Hz. Data were calculated by averaging 240 trials (80 trials for each class).

Figure 2 shows the time-frequency maps of the band power in the form of event-related desynchronisation/synchronisation (ERD/ERS) calculated using data from one subject. By visually checking these maps, we manually chose two bands for each channel as

features of three classes from this subject. Thus, there are ten features in total for the classifiers. It has been found that simple linear classifiers were just marginally worse than complex non-linear methods (Penny and Frost, 1996; Müller et al., 2003). It was shown in the BCI competitions 2003 and 2005 that LDA performs as well as (sometimes even outperforms) SVMs (Blankertz et al., 2004), and almost all the winning classifiers are linear (Blankertz et al., 2006). Therefore, LDA is chosen in our design. As LDA is not directly appropriate for three-class classification, we used three one-vs-rest binary LDA classifiers. Rifkin's analysis and review (Rifkin and Klautau, 2004) has shown that, for multi-class problems, the 'one-vs-rest' scheme can be as accurate as any other approaches.

Figure 1 The bipolar electrode positions used in our experiment (see online version for colours)

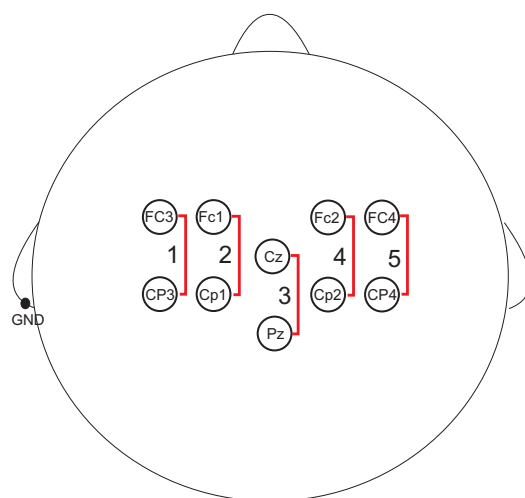
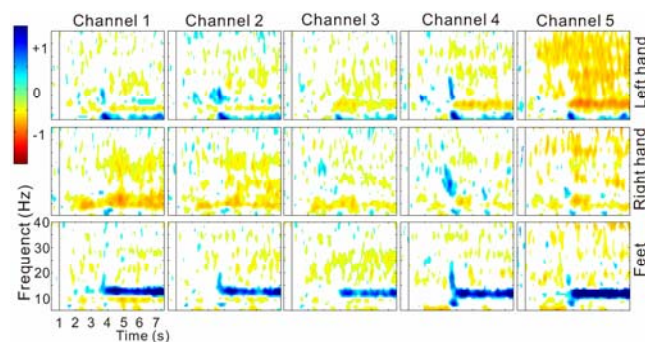
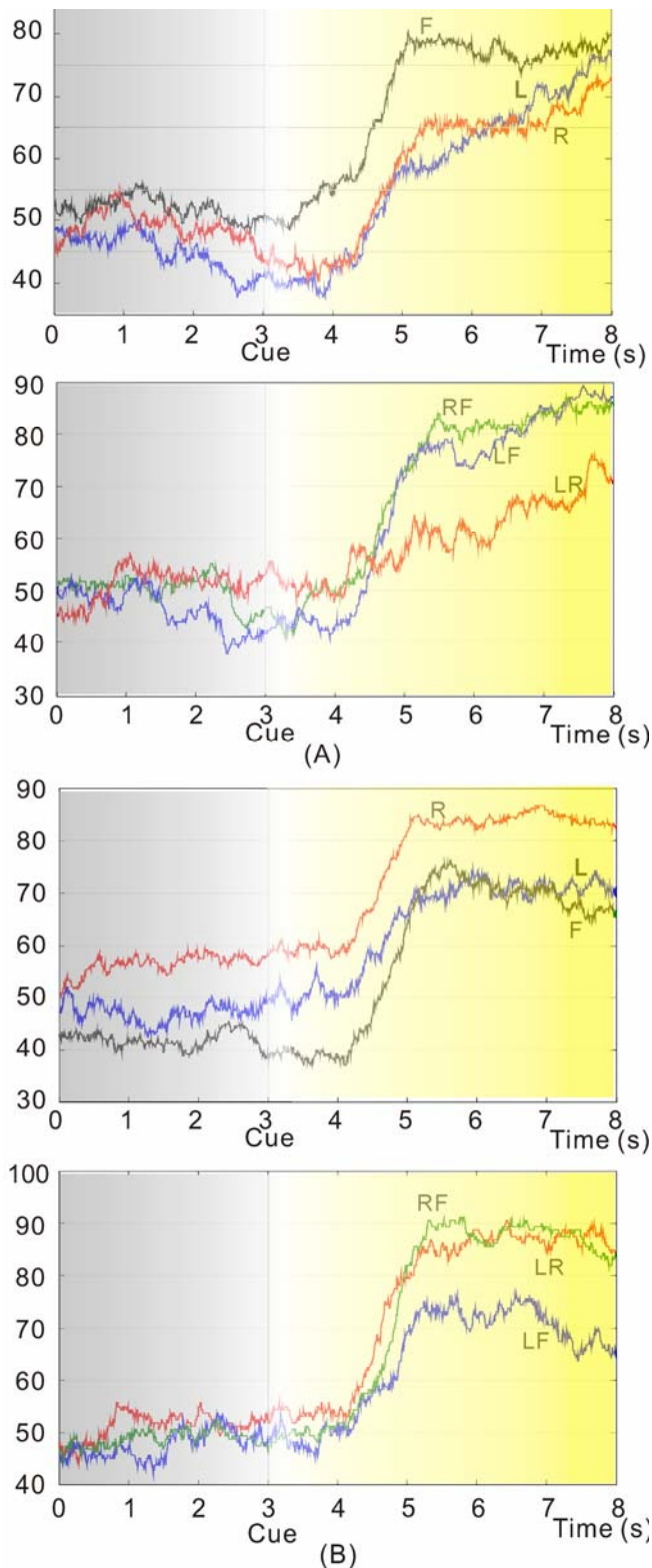


Figure 2 The time-frequency maps showing the ERD/ERS for subject 1 in terms of three motor imagery tasks, five bipolar channels (see online version for colours)



The LDA classifiers were trained with the offline data first, and then used in the online self-paced BCI. Figure 3 shows the accuracies of the classifiers validated using the leave-one-out method. The upper row shows the accuracies of three one-vs-rest LDA classifiers that will be used in the switching controller. LDA L is for left hand, LDA R for right hand, LDA F for feet. The lower row shows the performance of three pairs of classifiers for turning control. LR stands for left hand vs right hand, LF for left hand vs feet, RF for right hand vs feet.

Figure 3 Classification accuracies of the LDA classifiers of (A) subject 1 (B) subject 2 (see online version for colours)

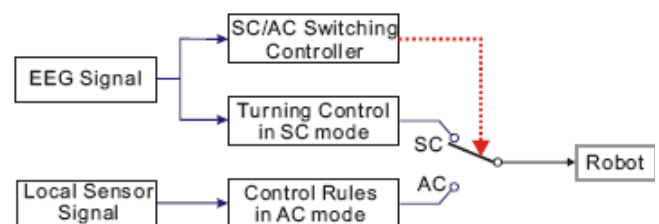


Note: The cue appears at 3 s.

3 Online self-paced BCI experiment

In the online self-paced BCI experiment, the subject sat in a comfortable arm chair, one metre away from a 19" screen on which a simulated robot and an indoor environment were displayed. The task of the subject is to control the robot to reach randomly chosen target locations in different rooms by imagining his/her hands and feet movements. Each spontaneous EEG sample of five channels was processed online with band power feature extraction and LDA classifiers. Combining the outputs of the LDA classifiers, the system used two controllers, i.e., a switching controller and a turning controller to control the movement of the robot, as shown in Figure 4. The system has two control modes: subject control (SC) and automatic control (AC). In the SC mode, the robot stops moving and the subject can turn the robot left or right by imagining left or right hand movements. When the system is in the AC mode, a group of rules triggered by local sensor signals will be active to control the robot movement. The switching controller detects the subject's motor imagery which is the most reliable, and switches the system between SC and AC modes. It has the highest priority and is always active. The turning controller is, instead, only active during the SC mode, controlling the turning movement. These two controllers and the rules will be described below in detail.

Figure 4 The control structure of the whole system (see online version for colours)



3.1 Switching control of SC/AC modes

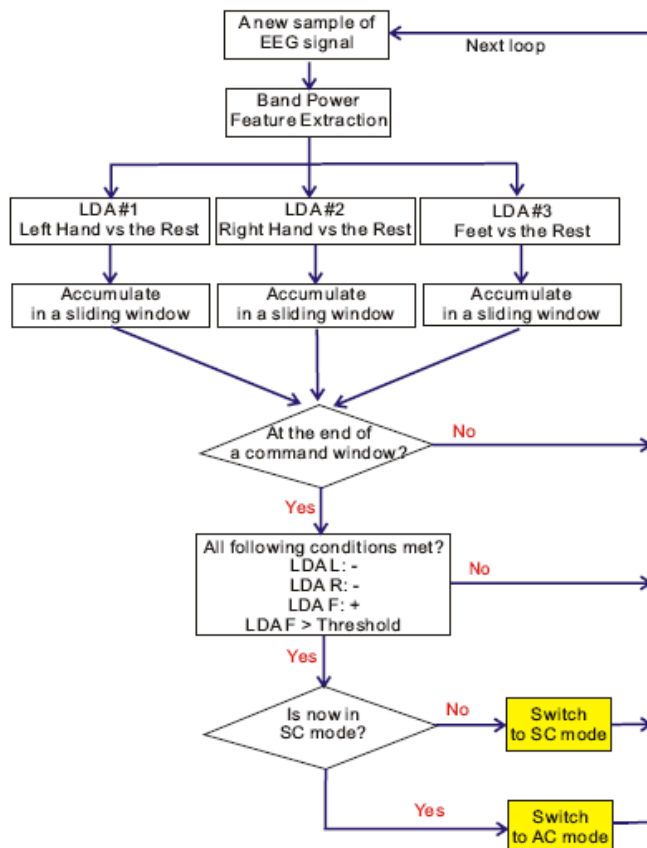
Since the switching controller is always active and has a higher priority over the turning controller and the AC control rules (see Figure 4), its reliability can affect the performance of the whole system greatly. Now the question raised is, which one among the three classes should be used for switching control. As shown in Figure 3, the classification of three motor imagery tasks has different reliability and accuracy in each subject. In subject 1, the accuracy of three classifiers involving feet is the most reliable class (see Figure 3A). In contrast, the right hand is the most reliable class in subject 2 (see Figure 3B). For this reason, we should not treat the three classes (mental tasks) equally as in other EEG-based BCIs (Millan et al., 2004; Rebsamen et al., 2007). For subject 1, whenever the motor imagery of feet is detected during SC or AC mode, it triggers mode switching. Meanwhile, the classification of the other two mental tasks, left hand and right hand, can only change the turning command during SC mode. But for subject 2, right hand is used for mode-switching control, and left hand and feet for left/right turning in SC mode.

Note that, subject 1 is taken as an example for simplicity from now on, using feet for mode-switching and left/right hands for turning.

The SC/AC switching controller is based on three one-vs-rest LDA classifiers (see Figure 5). The output of each LDA classifier is accumulated and averaged in a sliding window of one second. A new switching command will be generated at the end of each command window (one second) only if all the following five conditions are met.

- Condition 1: The output of LDA L (left hand) is negative.
- Condition 2: The output of LDA R (right hand) is negative.
- Condition 3: The output of LDA F (feet) is positive.
- Condition 4: The output of LDA F is higher than its threshold.
- Conditions 1–4 were not all satisfied at the end of the last command window.

Figure 5 The diagram of spontaneous EEG classification and the strategy for the mode-switching control (see online version for colours)



Note that, the one-second command window is not a sliding one. When one command window ends, the next command window begins. That means, the command sent to the robot is updated every one second. This strategy is inspired by our observation of the accuracy curves of the LDA classifiers obtained in the analysis of the synchronous experiment data

(see Figure 3). In each trial, the time period from 0 s to around 3 s (the left shadowed area in Figure 3) is similar to an idle state of a self-paced scenario, during which the accuracies of the three LDA classifiers are roughly at the chance rate, 0.5. Combining the three classifiers, we can have eight possible sets of outputs of the system (i.e., +++, ++-, +-+, +--, ---+, ---, ---). In the idle state, the probability of the three classifiers to produce an output of ‘---’ is roughly at 1/8. That means the probability of mistakenly recognising an idle state (AC mode) as a switching command will be quite low. Due to the effect of the given threshold of LDA F in condition 4 above, this probability can be further reduced to even lower than 1/8. On the other hand, when the subject is doing the motor imagery of feet movements (see the right shadowed area in Figure 3) because the accuracy of the three classifiers are quite high during the period starting after 4 s, the probability of producing an output of ‘---’ can also be quite high. With this rule in the control mode, the system will be more likely to make a right decision when the subject intends to switch the robot’s moving mode.

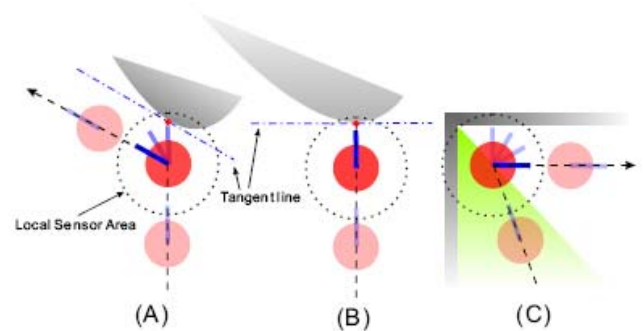
Condition 5 is to avoid the unintended repetitive switching of SC and AC modes in two consecutive command windows, which could happen easily as the command window is one second while the EEG pattern of a motor imagery could last for more than one second.

3.2 Control rules in the AC mode

The robot is be equipped with sensors that can detect the form of any objects in a small local area around it, as shown in Figure 6, but all other objects in the environment outside its local sensor area are invisible to the robot.

- *Obstacle-rule-1*: Follow the tangent line of the obstacle.
- *Obstacle-rule-2*: Stop moving and switch the system mode to SC if facing up to an obstacle at 90°.
- *Corner-rule*: To manoeuvre in a corner.

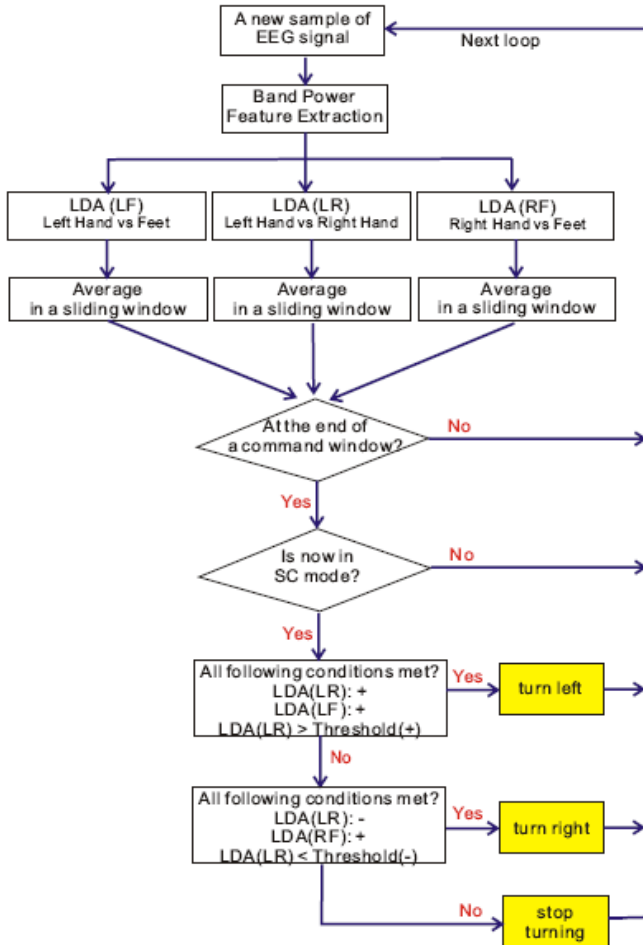
Figure 6 The control rules of the AC mode (see online version for colours)



During the AC mode, the robot always automatically keeps moving forward at its orientation until one of the following two events are detected by its onboard sensors:

- When the robot senses an obstacle/wall in its moving direction in the local sensor area, it calculates the tangent line at the point of the obstacle it is approaching, as shown in Figures 6(A) and 6(B). If the angle between this tangent line and the moving direction of the robot is not at around 90° , the robot will automatically turn by an angle of less than 90° and continue moving forward following the direction of this tangent line, as shown in Figure 6(A). We call this ‘obstacle-rule-1’. If this angle is at around 90° , the robot will stop at its current position and heading, and the system will be changed to the SC mode, allowing the subject to take over its control, as shown in Figure 6(B). This rule is named ‘obstacle-rule-2’.
- If the robot detects a corner, it will turn its direction in the way as illustrated in Figure 6(C) and move out of the corner and then forward. This rule is called ‘corner-rule’. The corner is divided into two areas by its middle line and the past path of the robot is supposed to be in one area (the shadowed one). To move out the corner, the robot automatically turns and follows the wall in the other area.

Figure 7 The diagram of spontaneous EEG classification and the strategy for turning control in the SC mode (see online version for colours)



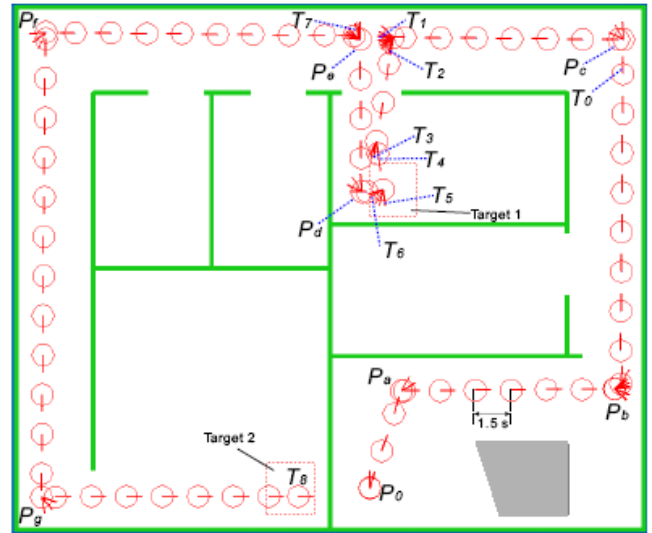
3.3 Turning control in the SC mode

Once the system is switched to the SC mode, the robot immediately stops its moving and the turning controller takes control to change the robot’s heading based on the EEG signals that are processed in the same way as in the switching controller to reduce the probability of error classification as shown in Figure 7. However, three LDA classifiers are different because the turning controller only requires a two-class classification for generating the command of turning left or right. Now the three classifiers are LR (left hand vs right hand), LF (left hand vs feet) and RF (right hand vs feet). The conditions for generating turning commands are shown in Figure 7. For example, if the outputs of both LR and LF are positive and the output of LR is over the threshold, the robot will turn left.

4 Experimental results

Three subjects selected took part in our self-paced BCI experiments. Subjects 1 and 3 are well-experienced in BCI experiments. Subject 2 is naive, having no experience in BCI experiments. The simulated indoor environment is composed of six rooms and three corridors (see Figure 8), including some obstacles and specified targets in the rooms. The robot moves at a speed of 1.4 times of its width per second, and turns at a speed of 20° per second.

Figure 8 A series of snapshots of the robot in an experiment by subject 1 (see online version for colours)



In Figure 8, the time interval between any two successive snapshots is 1.5 second. The task in this experiment was to reach the two targets in two separate rooms. At the beginning of the experiment, the robot was placed at P_0 and the system was in the SC mode allowing the subject to control its turning, as shown in Figure 8. The subject turned the robot to right by imagining his or her right hand and then switched the system to the AC mode by imagining his/her feet.

In the AC mode, the robot keeps moving forward and the subject can have a rest. At P_a , the onboard sensor detected a wall and, due to the effect of an AC rule, ‘obstacle-rule-1’ in Figure 6(A), the robot automatically turned right to follow the wall. At P_b , the robot is facing up another wall nearby. With the effect of another AC rule, ‘obstacle-rule-2’, the robot stopped and the system was switched to SC mode. With the target in mind, the subject tried to change the heading of the robot to left by imagining left hand movement, but the turning controller misclassified the subject’s intention and the robot was turned right first.

The system then correctly classified the subject’s mental state and the robot turned to left direction. The subject changed the system to AC mode immediately by imagining his feet and the robot started moving forward. With the effect of ‘obstacle-rule-1’ again, the robot automatically turned to the direction of the wall near P_b and followed it until arriving at P_c , at which a corner was detected and the robot automatically turned left by 90° and continued moving forward according to one of the AC rules, ‘corner-rule’ in Figure 6(C).

At T_1 , the system classified the subject’s motor imagery of feet and the system mode is switched to SC, allowing the turning controller to take effect. After a few mistaken right-turning, the robot was turned to an appropriate left-orientation (T_2) by the subject’s repetitive imagination of left hand in several seconds. Then the subject changed the system to AC to start the robot by imagining his/her feet and the robot successfully entered the door. At T_3 , the subject stopped the robot by changing the system to SC mode via motor imagery of the feet, and turned the robot to the direction of Target 1 in the room, then started it again at T_4 by switching the system to AC mode using feet imagery. At T_5 , the robot reached Target 1 and the subject stopped it and turned it right to move out of the room. Likewise, in the journey for Target 2, the mode of the system was switched at T_6 , T_7 by the classified subject’s feet motor-imagery, and at P_e by the AC control rule, ‘obstacle-rule-1’. After following the corners P_d , P_f and P_g , the robot hit Target 2 in another room. It may worth noting that the time period from T_7 to T_8 is a long-time AC mode of 60 seconds, during which the robot is totally under the control of the AC rules without any intention from the subject.

In this experimental run, while the switching controller had not produced any wrong commands during the long periods of AC mode, the turning controller based on the left/right hand classification in the SC mode made many mistakes (see the wrong turnings at P_b and T_1 in Figure 8). The reason for this is, for subject 1, the accuracy and reliability of the discrimination between left and right hands is low (see Figure 3). However, due to the intuitive nature of the switching/turning controllers and the simplicity of the AC rules, the subject can directly see what mental state has been recognised in the controller by observing the movement of the robot. As shown in Figure 8, the subject had been able to correct these turning errors by enhancing his motor imagery of left/right hands because, in SC mode, the robot keeps stopping and only allows turning control.

However, these corrections are time consuming sometimes, which can be seen more clearly in a video clip of this experiment at <http://dces.essex.ac.uk/staff/tgeng/asyn>.

Figure 9 A segment of the outputs (averaged in a sliding window) of the three one-vs-rest LDA classifiers used in the switching controller (see Figure 5), (A) feet (B) left hand (C) right hand (see online version for colours)

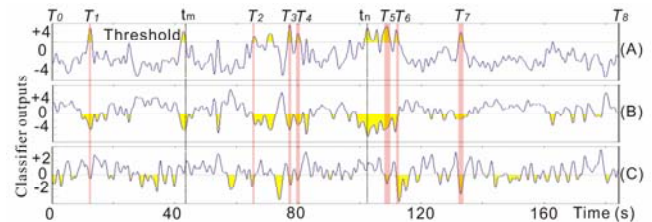
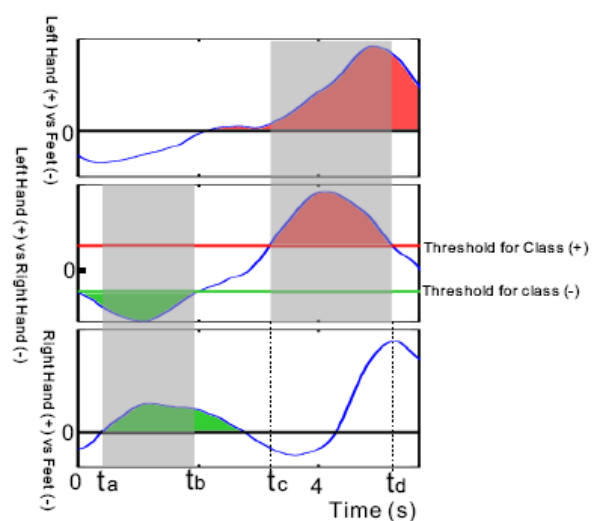


Figure 9 is real-time data recorded in the experiment shown in Figure 8. The time length is 185 seconds, from T_0 to T_8 . The time indicated with T_0 , T_1 , T_2, \dots, T_8 are corresponding to those positions of the robot labelled with T_0 , T_1 , T_2, \dots, T_8 in Figure 8. During the seven short time periods T_1 , T_2, \dots, T_7 (the shadowed areas), the five conditions of the switching controller shown in Figure 5 are met and the mode of the system is switched. During all other time periods between them, the five conditions of the switching controller are not all satisfied, and thus have not led to any mode-switching. For example, at time t_m and t_n , the output of the classifier for feet is over the threshold and the output of the classifier for left hand is negative, which satisfy the first, the second and the fourth condition of the switching controller, but the output of the classifier for right hand is positive, not matching the third condition.

Figure 10 A segment of the outputs (averaged in a sliding window) of the three LDA classifiers used in the turning controller in SC mode (see Figure 7), (A) LF (B) LR (C) RF (see online version for colours)

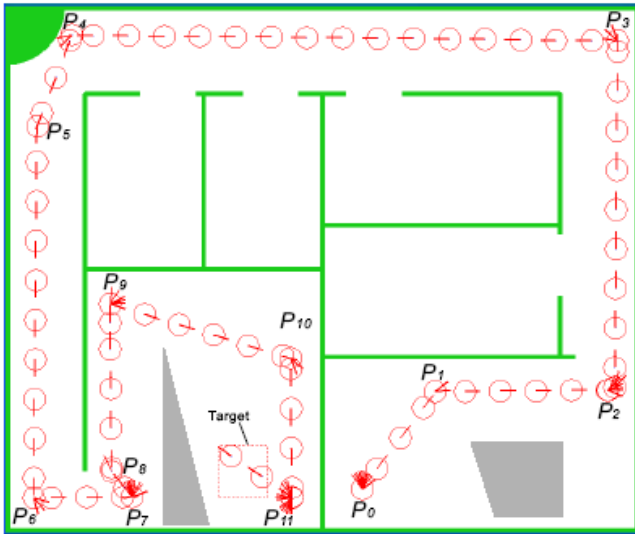


As an example of turning control in SC mode, Figure 10 shows the outputs of three classifiers used in the turning controller in SC mode corresponding to the time slot between T_3 and T_4 in the experiment shown in Figure 8. In

Figure 10, the data is recorded in the experiment, corresponding to the time slot of SC mode between T_3 and T_4 in Figure 8. From t_a to t_b , the conditions for right turning were met, the robot turned right during this period. Similarly, the robot turned left in the period between t_c and t_d as the condition for left turning was satisfied in that time.

In another run, the task is to reach a target behind a big obstacle (see Figure 11). After being turned and started by subject's intentions at P_2 , the system kept in AC mode for 78 seconds before arriving at P_7 , and revealed no false positive in this period. At P_7 , the system was switched to SC mode by the AC control rule and the subject turned the robot to left first and then started it. At P_8 , the robot automatically turned to follow the wall under the control of the AC rule, 'obstacle-rule-1'. At P_9 , the subject stopped it by switching to SC mode. After being turned to the direction of the target, the system was switched to another AC mode which lasted until P_{11} . At P_{11} , the subject stopped the robot and turned it right, then controls it to reach to the target.

Figure 11 A serials of snapshots of the robot in another experiment in which the subject tried to control the robot to circumnavigate a big obstacle and reach the target (see online version for colours)



In order to evaluate the performance of the BCI system, three subjects were trained for two sessions (one hour per session). Eight runs conducted were recorded for evaluation, in which the time-length of each trial is various due to the different performance of subjects and the various degree of difficulties of the tasks (see Table 1).

It is difficult to estimate accurately the subjects' real intentions at every moment of a self-paced online experiment. The accuracy of SC mode in our experiments is calculated in a way similar to those methods excluding the unknown state (Millan et al., 2004). For example, in Figure 8, the time period (T_1 - T_2) is an SC mode with a time-length of 53 seconds (see Figure 9), of which there are 53 commands generated. Since Target 1 is at the left side of the robot at this moment, the subject's intention is always to turn left. Therefore, 13 commands are 'turning left', nine are

'turning right', 31 are 'stopping'. The stopping command here is generated when the turning controller cannot make decision on 'left' or 'right' (see Figure 10), which is similar to the 'unknown' state in Millan et al. (2004). The accuracy rate of this period is calculated as, $13/22 = 0.55$.

As shown in the experiments above, an AC mode can be quite long, during which the system could mistakenly recognise the subject's mental state as feet imagination and thus wrongly switch to the SC mode. For this reason, we used the error rate to evaluate the performance in the AC mode. As shown in Table 1, the accuracy of subject 1 in SC mode is shown low. This is because the accuracies of the classifiers in subject 1 [see Figure 3(B) lower] used for turning control in IC mode is much lower than that of subject 2 [see Figure 3(A) lower].

We used a simulated robot in the online experiments above because it is convenient for recording the movement and analysing the BCI performance. Subject 2 has also participated in an experiment controlling a real NXT mobile robot using the same control structure. From the video footage, <http://dces.essex.ac.uk/staff/tgeng/nxt.avi>, it can be seen that the performance of real robot control is similar to that of the simulated robot control.

Table 1 Results of eight runs from three subjects

Subject	Runs	Time-length of the run (s)	Time-length of AC modes (s)	Time-length of SC modes (s)	Error rate of AC mode	Accuracy of SC mode
1	1	257s	152s	125s	2.0%	68%
	2	381s	126s	255s	0%	71%
	3	288s	117s	171s	0%	66%
2	1	295s	168s	127s	2.9%	86%
	2	264s	138s	126s	3.0%	82%
	3	317s	139s	178s	1.4%	87%
3	1	286s	129s	157s	1.4%	89%
	2	364s	131s	233s	0.8%	85%

5 Conclusions and future work

In this paper, two strategies have been proposed for self-paced online BCI control of a simulated robot, i.e., using the most reliable mental task in a subject for switching control of the SC/AC mode, and combining three binary classifiers to make decisions. The design of the controller structure has two main characteristics. First, it has a hierarchical structure, allowing the most reliable mental task to play a dominant role in the robot control. Second, its intuitive nature can provide straightforward feedback to the subject. The subject can easily know the state of the controller by only observing the current movement of the robot.

However, the intuitive controller in our system has a limitation, i.e., frustration and fatigues sometimes caused by the mistaken turning commands in the SC mode, and the

mental burden in the subject when dealing with complex manoeuvres such as entering a door (see the experiment shown in Figure 8 and its video). As demonstrated in the shared-control structures of joystick-driven or voice-driven wheelchairs (Yu et al., 2003; Pires and Nunes, 2002), it could be helpful to use an autonomous controller in EEG-controlled wheelchair for difficult circumstances. Due to the uncertainty and extremely low data-rate of EEG-based control, there is a dilemma between autonomy in the device and flexibility in the user's control. In a BCI system with more autonomy, an error in sensor signals or a misclassification of user's mental states could cause a series of wrong movements or responses in the controlled device, and make the user feel the device is out of control. How to balance between these two aspects in EEG-controlled robots will be our future work.

References

- Blankertz, B., Müller, K., Curio, G., Vaughan, T., Schalk, G., Wolpaw, J., Schlogl, A., Neuper, C., Pfurtscheller, G., Hinterberger, T. et al. (2004) 'The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials', *IEEE Transactions on Biomedical Engineering*, Vol. 51, No. 6, pp.1044–1051.
- Blankertz, B., Müller, K., Krusienski, D., Schalk, G., Wolpaw, J., Schlogl, A., Pfurtscheller, G., J.R., M., Schroder, M. and Birbaumer, N. (2006) 'The BCI competition III: validating alternative approaches to actual BCI problems', *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 14, No. 2, pp.153–159.
- Borisoff, J., Mason, S. and Birch, G. (2006) 'Brain interface research for asynchronous control applications', *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 14, No. 2, pp.160–164.
- Halder, B. and Sarkar, N. (2007) 'Robust nonlinear analytic redundancy for fault detection and isolation in mobile robot', *International Journal of Automation and Computing*, Vol. 4, No. 2, p.177.
- Millan, J., Renkens, F., Mourino, J. and Gerstner, W. (2004) 'Noninvasive brain-actuated control of a mobile robot by human EEG', *IEEE Transactions on Biomedical Engineering*, Vol. 51, No. 6, pp.1026–1033.
- Müller, K., Anderson, C. and Birch, G. (2003) 'Linear and nonlinear methods for brain-computer interfaces', *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 11, No. 2, pp.165–172.
- Oya, M., Tsuchida, Y., Wang, Q. and Taira, Y. (2008) 'Adaptive active suspension controller achieving the best ride comfort at any specified location on vehicles with parameter uncertainties', *International Journal of Advanced Mechatronic Systems*, Vol. 1, No. 2, pp.125–136.
- Penny, W. and Frost, D. (1996) 'Neural networks in clinical medicine', *Med Decis Making*, Vol. 16, No. 4, pp.386–98.
- Pires, G. and Nunes, U. (2002) 'A wheelchair steered through voice commands and assisted by a reactive fuzzy-logic controller', *Journal of Intelligent and Robotic Systems*, Vol. 34, No. 3, pp.301–314.
- Rebsamen, B., Burdet, E., Guan, C., Zhang, H., Teo, C., Zeng, Q., Laugier, C. and Ang, M. Jr. (2007) 'Controlling a wheelchair indoors using thought', *IEEE Intelligent Systems*, Vol. 22, No. 2, pp.18–24.
- Rifkin, R. and Klautau, A. (2004) 'In defense of one-vs-all classification', *The Journal of Machine Learning Research*, Vol. 5, pp.101–141.
- Sato, N., Matsuno, F. and Shiroma, N. (2008) 'Development of a high mobility wheeled rescue robot with a 1-DOF arm', *International Journal of Advanced Mechatronic Systems*, Vol. 1, No. 1, pp.10–23.
- Tanaka, K., Matsunaga, K. and Wang, H. (2005) 'Electroencephalogram-based control of an electric wheelchair', *IEEE Transactions on Robotics*, Vol. 21, No. 4, pp.762–766.
- Wu, C., Zhang, Y., Li, M. and Yue, Y. (2006) 'A rough set GA-based hybrid method for robot path planning', *International Journal of Automation and Computing*, Vol. 3, No. 1, pp.29–34.
- Yu, H., Spenko, M. and Dubowsky, S. (2003) 'An adaptive shared control system for an intelligent mobility aid for the elderly', *Autonomous Robots*, Vol. 15, No. 1, pp.53–66.