# Brainware: Synergizing Software Systems and Neural Inputs

Shihong Huang
Computer & Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton, FL 33431, USA
+1-561-297-1275
shihong@fau.edu

Emmanuelle Tognoli
Center for Complex Systems & Brain Sciences
Florida Atlantic University
Boca Raton, FL 33431, USA
+1-561-297-0110
tognoli@ccs.fau.edu

## ABSTRACT

The rapid advances in the field of Brain Computer Interfaces (BCI) are expected to enrich the quality of people's lives. BCI connects computer actions with neural inputs—signals indicating the user's intentions, desired actions, attention, thoughts, memories, and emotions. BCI applications present significant challenges for computer science and software engineering research: an avalanche of neural signals will make their way as direct input into software systems. Given the differences between neural inputs and behavioral ones, the integration of neural inputs will require special approaches, and not simply adding yet more user interface channels to pre-existing software systems. This paper explores the challenges of designing and implementing self-adaptive software systems that could synergize brain states. After framing the problem, its rationale and possible solutions, in this paper we argue that the software engineering community ought to investigate how to incorporate neural inputs into software systems. The days are now upon us when software systems can "feel" and "anticipate" the users' intentions and therefore react self-adaptively and synergistically to their needs.

## Categories and Subject Descriptors

D.2.2 **[Design Tools and Techniques]:** User interfaces

H.1.2 **[User/Machine Systems]:** Human factors

## General Terms

Human Factors, Performance

## Keywords

Brain computer interface (BCI), human computer interface (HCI), neural input, self-adaptive systems, overt and covert behavior and attention, human in the loop



*ICSE'14*, May 31–June 7, 2014, Hyderabad, India.
Copyright 2014 ACM 978-1-4503-2768-8/14/05... $15.00...$15.00.

## 1. INTRODUCTION

Humans are highly evolved and adapted organisms who have achieved a high level of intelligence and efficiency in cognition (information processing) and behavior (enactment). Computers have both inspired and been inspired by human information processing [28][7]. The landmark design of von Neumann's stored-program architecture was based on performance of arithmetic calculations and information processing by humans and designed as a fast and automatic surrogate for such tasks. In turn, the digital computer has become, over the course of many decades, a highly influential model for understanding and interpreting human brains and behavior, as evidenced by the lasting success of cognitive neuroscience.

This long history of reciprocal inspiration between brains and computers has captured human imagination. Many sci-fi movies foresee a future where computer systems are controlled by one's brain. In turn, a software system could behave as an artificial brain that can "feel" and "anticipate" human intentions and therefore react self-adaptively to its needs. The advancement of BCI techniques and availability of EEG to the general population is making this dream come true. The applications of BCI include medical (e.g., sensory and motor restoration [2]) as well as nonmedical applications (e.g., navigating virtual environments and Google earth [25]). The seamless integration of humans and computers in our daily lives and the high demand for interaction efficiency will likely require a dramatic change or even a paradigm shift in computer science and software engineering to accommodate neural signals in software systems.

In this paper we advocate that the time has come to embrace the challenges and to underline the importance of covert behaviors (i.e., behaviors that are not accessible to the casual observer, such as covert shift of attention, memory retrieval, perceptual integration, planning, or action preparation) that will go beyond the more familiar overt user behavior (e.g., mouse, keyboard, or voice commands). The rest of this paper is organized as follows. Section 2 illustrates the need to augment computing systems with neural inputs. Section 3 describes enabling technologies and BCI types to realize such goals. Section 4 presents neural inputs as the missing link in self-adaptive systems. Finally, Section 5 concludes the paper and outlines avenues for research.

## 2. FROM BEHAVIOR TO BRAIN INPUT

### 2.1 Bottleneck of HCI: Behavioral Inputs

Brains are massively parallel systems, performing numerous processes such as perception, logical reasoning, action planning, memory retrieval, support of the body's physiological functions, attentional orienting, all at once (cf. Fig. 1.1). On the other end of the interaction, computers are also massively parallel information processing systems (cf. Fig. 1.3-7). Between brain and computer stands the bottleneck of Human Computer Interaction (HCI): the overt human behavior (cf. Fig. 1.2) channeled to the computer by attached input devices (cf. Fig. 1.3). User inputs are mapped into computer actions (cf. Fig. 1.5-6) by a series of software instructions (cf. Fig. 1.4). The outputs (i.e., sound notifications and screen refresh) are returned to the user through perception (cf. Fig. 1.7), eliciting a new set of brain states, functionally translated into covert human behavior (cf. Fig. 1.8), and ultimately leading to a new selection of overt behaviors (cf. Fig. 1.1). Human behavior is notoriously slow and essentially serial, with only a selected few patterns of multitasking found to be stable either inherently or after training has led to the creation of attractors in the behavior's coordination dynamics [16]. Human behavior is the weak link that limits the performance of HCIs. Consequently, this bottleneck prevents software developers to push speed and parallelism beyond the normal rate of information encountered in human behavior.

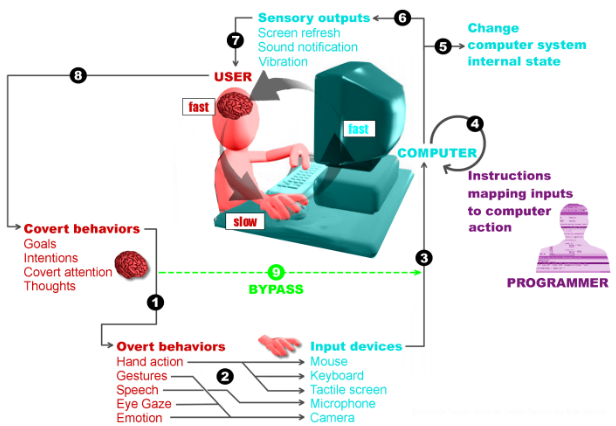


**Figure 1: An outline of information flows in contemporary computer-interaction systems (1-8) and future prospects of integrating covert human behavior decoded from neural signals (9).**

## 2.2 Forthcoming Neural Inputs

One contemporary development that could transform the cooperation between humans and machines, and fundamentally alter the demand on software systems, is the generalized incorporation of neural inputs into computer systems. Neural inputs are decoded information about users' brain states and their corresponding covert behaviors. The neural signals that convey those states can be acquired through suitable sensor arrays, decoded and directly addressed to the computer (cf. Fig. 1.9) where they trigger specific actions (cf. Fig. 1.5-6) according to software instructions (cf. Fig. 1.4) in the absence of any overt behavior (cf. Fig. 1.2). Such neural inputs provide a fast connection between users' mental processing (e.g., intentions) and the processing performed by the computer. To the present day, software engineering has had few if any direct neural inputs to deal with, but this situation is about to change. From the viewpoint of a computer system, neural inputs are different from behavioral ones. Covert human behaviors such as intention, attention, and thoughts occur at a faster rate and significantly earlier than the overt behavior that they elicit (e.g., action, orientation, communication, or gaze). Covert behaviors also tend to happen, not as a nicely organized series of steps, but rather as an integrated ensemble that can come and go, and reconfigure its associations according to demand. Those properties require a fundamental change in the way inputs are dealt with by software systems.

# 3. ENABLING TECHNOLOGIES: BCI

Electroencephalography (EEG) is the most common BCI enabling technology, which measures ionic current flows in the brain's neurons over the course of brain functions. This technique recently evolved from specialized equipment reserved for neuroscientists to more accessible gadgets for ordinary users [10]. This move was made possible by innovations that evolved from "wet" electrodes operated with conductive gel, expensive hardware, high-end bioamplifiers and high-density wiring, to consumer-use EEGs that use dry electrodes and wearable headsets with lightly-wired and even wireless transmission of brain signals [11][8][29][19].

## 3.1 Principles of BCI

Today the minimal BCI system includes: sensing equipment to measure brain activity, a decoder, interpreter or classifier to link the measured brain activity with specific brain states (e.g., pattern recognition algorithms, a support vector machine, independent or principal component analysis (ICA/PCA), as well as artificial neural networks), and an output that changes the state of the computer system according to rules of the BCI (e.g., *if attention positioned on email application icon, prepare launch*). Usually, the BCI functions are in a closed loop, which means that the agent whose brain activity is interfaced is made aware of the changes that have occurred as a result of her brain activity. A variety of neuroscience techniques are leveraged to measure brain activity, including EEG, Magnetoencephalography (MEG), the slower and non-portable functional Magnetic Resonance Imaging (fMRI), as well as functional near-infrared spectroscopy (fNIRS). All of these techniques have been explored for BCI, but EEG has a clear advantage for consumer-aimed BCI applications, due to speed, low-cost, and portability. EEG-based BCI are further subdivided into invasive (e.g., human epileptic patients prepared for brain surgery), semi-invasive BCIs, and non-invasive BCIs [18]. Invasive recording techniques allow the activities of single neurons or populations of neurons to be recorded with electrodes implanted into the brain. The recorded activity is most precise and detailed, but the technique is limited to people with neurosurgical conditions and a priori excludes the normal population. For the latter, the only option is non-invasive techniques. A significant challenge is to translate recorded neural activities into information about the user's mental state.

## 3.2 BCI Examples

Early examples of invasive BCIs in animals demonstrated neural control of a robotic arm by rats [5] and monkeys [4]. BCI-controlled movements have evolved from one to three dimensions [24], and recently, a monkey successfully used a prosthetic arm to feed itself [26]. For non-invasive EEG, pioneering research uses brain waves and a computer to spell letters and communicate messages involving patients who had totally lost the capability to communicate [12][23]. This type of BCI resorted to selective attention to identify which letter displayed on a screen a person was attentive to. A number of applications also examined how to control a cursor or an object on the screen using brain activity [31][2] or objects placed in the environment [21]. Interest also arose for signals encoding the brain's emotional states through the development of effective BCIs [20].

## 3.3 Preliminary Work on Covert Attention

Our own research group has developed a BCI prototype based on *covert attentional shifts* [3]. It exploits the neuromarker ξ that arises in the brain when users seek information in the periphery of the visual field. Our ongoing research aims to decode the position of the user's focus of attention from brain signals. If successful, with additional knowledge of the position of the user's head and screen, it would be possible to map the covert attention trajectory in real time. In particular, when subjects interact with computers, it may become possible to determine which icon or parts of the screen the user is paying attention to (while eye gaze is focused on a different location). This information opens up possibilities to develop a computer system endowed with predictive capabilities. It provides contingencies for the future actions that the user is likely to undertake in the next 0.25-3.0s. Accordingly, it could be used to pre-emptively pre-allocate computer resources and background tasks to fulfill the user's intended action.

## 3.4 Opportunities

As our partial overview shows, the BCI field comprises many research projects in the neuroscience community. Moreover, several BCIs have been successfully implemented—dealing with selective attention, motor preparation, and motor control. The software engineering community should seize this unique opportunity and investigate how to capture and exploit neural inputs in software systems.

# 4. SELF-ADAPTIVE SYSTEMS

From a software engineering perspective, the increasing complexity and dynamism of software intensive systems require that the systems be able to adapt themselves at runtime to react and deal with the uncertainties and the unpredictable nature of the environments in which they operate [13]. The systems should be able to adapt to different user needs, with different resources, intrusions, faults and exceptions. The self-adaptive capability requires the system to be able to modify its behavior and/or structure in response to its perception of the environment, the system itself as well as its goals [9].

For the past decade, much research has been undertaken in developing methods, techniques, and tools to address the needs of software-intensive systems that are self-adaptive in nature [6]. For example, defining models that can represent a wide range of system properties [1], managing uncertainty at the requirements stage [30], making feedback control loops more explicit [17], building architecture based self-adaptation [14][15], and creating frameworks for assessing and certifying adaptation properties of self-adaptive systems [27].

Self-adaptive systems research has focused on systems' reaction to resources, environments, exceptions, and recovery. Humans constantly interact with the physical world [22]. Human-in-the-loop and covert behavior should be recognized when systems make decisions and react accordingly.

## 4.1 Missing Link: Human Neural Inputs

Self-adaptive software systems feed on contextual information from the environment, from within themselves, and from the user. A wealth of data about the user's intentions, thoughts, emotions, and desires is available, but hidden in the patterns of her brain (i.e., covert behaviors). Therefore, one of the missing links in the many existing self-* properties of self-adaptive systems is to address how to anticipate and react to human thoughts and mental states, which can be accomplished by taking neural inputs into account. Neural inputs provide ample opportunities to incorporate human-in-the-loop mechanisms into self-adaptive systems. This constitutes a paradigm shift in software systems—from reacting to anticipating.

## 4.2 The Challenges of Integrating Neural Inputs into Software Systems

When facing this pending arrival of neural inputs in software systems, many questions arise: How do we integrate this wealth of new inputs into existing software architectures? How do we recognize covert behaviors to provide better experiences to end-users? How do we coordinate and correlate neural inputs with behavioral inputs and with each other to ensure that they will facilitate rather than interfere with human computer interaction? We propose to investigate these research questions by characterizing appropriate self-adaptive systems. Neural input is a natural place to start for a number of reasons. There are similarities in functional architectures, especially the pervasive existence of feedback loops, in both the brain and self-adaptive software.

Other auspicious parallels concern time effectiveness, autonomy, and the faculty to adapt to unpredictable circumstances. The self-adaptive systems research community has developed many mechanisms to cope and deal with uncertainty by adapting software systems at runtime. In general, uncertainty may be due to changes in the operational environment, variability of resources, and new user needs. As Garlan [13] posited, human behavior contributes large amounts of uncertainty, and if we add *covert* human behavior (e.g., intentions, desired actions, attention, thoughts, memories, or emotions) even more uncertainty will result.

The challenges of brain inspired self-adaptive systems are that the system should adapt to the uncertainty and transient covert behavior of the human brain and adapt effectively to user needs. Furthermore, it may use covert behavior inputs to reflect on its own response to earlier input.

# 5. CONCLUSION AND FUTURE WORK

This paper posited to the software engineering community that human covert behavior should be placed *in the loop.* We argued that direct connections between neural inputs and computer systems are inevitable and that the community needs to face these connections. Through the unique features of neural inputs (e.g., fast and parallel insight into user's intentions, desired actions, attention, thoughts, memories, or emotions), smart software systems will someday "feel" and "anticipate" users' intention and therefore react self-adaptively. To reach this goal, our current efforts are to list the properties that software architects need to consider to adapt to the neural inputs' specific properties. This list is inspired by the ways brains self-organize and integrate their many functions continuously over time. Our research creates a framework to bring neural signals into the realm of HCI and endow the cooperation between human and machine with features such as preparatory attention and intention. With those considerations in mind, we expect that this work will elevate software systems to a new level.

In time, this research will extend beyond the interaction between users and computing systems. The class of domains that would benefit most from direct neural inputs is where users and systems are intimately intertwined. For example, consider a smart home where the aim is to make the user happy. Today's smart environments are lacking largely because the system has so little knowledge about what the user wants, thinks, likes, or dislikes (i.e., human covert behaviors). As a result, it has been difficult to build effective systems that anticipate and react to user needs. But such immersive systems will proliferate as BCI technology emerges and software systems become more task-centric and responsive to context changes. In the self-adaptation realm, such systems are often referred to as *mixed initiative systems,* where users and the system work together closely to achieve common goals. Making use of the brain's highly efficient parallel, multifunctional processing and self-organizing functionalities, and having direct neural inputs to the systems is the natural next step to enable software systems to "feel" and "anticipate" users' intentions and thus react self-adaptively.

## 6. REFERENCES


[1] Andersson, J., et al. (2009). Modeling dimensions of self-adaptive software systems, in: *Software Engineering for Self-Adaptive Systems,* LNCS 5525:27-47.

[2] Birbaumer, N. (2006). Breaking the silence: Brain–Computer Interfaces (BCI) for communication and motor control, *Psychophysiology* 43(6):517-532.

[3] Calderon, R. (2007). *Brain Computer Interface and Neuroprosthetic*s, M.Sc. Thesis, Florida Atlantic University.

[4] Carmena, J.M., et al. (2003). Learning to control a brain-machine interface for reaching and grasping by primates, *PLOS Biology* 1(2):e42.

[5] Chapin, J.K., et al. (1999). Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Natural Neuroscience* 2(7):664-70.

[6] Cheng, B.H.C., et al. (2009). Software engineering for self-adaptive systems: A research roadmap, in: *Software Engineering for Self-Adaptive Systems,* LNCS 5525:1-26.

[7] Churchland, P.S., Sejnowski, T.J. (1992). *The Computational Brain*, MIT Press.

[8] Cognionics 64-channel wireless EEG Headset, retrieved Feb. 2014. www.cognionics.com

[9] de Lemos, R., et al. (2013). Software engineering for self-adaptive systems: A second research roadmap, in: *Software Engineering for Self-Adaptive Systems II,* LNCS 7475:1-32.

[10] Debener, S., et al. (2012). How about taking a low-cost, small, and wireless EEG for a walk? *Psychophysiology* 49(11):1617-1621.

[11] EPOC, retrieved Feb. 2014. emotiv.com/epoc/features.php

[12] Farwell, L.A., Donchin, E. (1988). Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potential, *Electroencephalography and Clinical Neurophysiology* 70(6):510–523.

[13] Garlan, D. (2013). A 10-year perspective on software engineering self-adaptive systems (Keynote), *Proc. 8th Int. Symp. on Software Engineering for Adaptive and Self-Managing Systems (SEAMS 2013),* p. 2.

[14] Garlan, D., et al. (2012). Foundations and tools for end-user architecting, in: *Large-Scale Complex IT Systems,* LNCS 7539:157-182.

[15] Garlan, D., et al. (2009). Architecture-based self-adaptation, in: *Autonomic Comp and Networking*, pp. 31-55.

[16] Kelso, J. A.S., Dumas, G., Tognoli, E. (2013). Outline of a general theory of behavior and brain coordination. *Neural Networks* 37:120–131.

[17] Müller, H., Pezzè, M., Shaw, M. (2008). Visibility of control in adaptive systems, *2nd Int. Workshop on Ultra-Large-Scale Software-Intensive Systems (ULSSIS 2008),* pp. 23-26.

[18] Rao, R.P.N. (2013). *Brain-Computer Interfacing: An Introduction.* Cambridge University Press.

[19] ReThink Medical Next Generation Wireless EEG, retrieved Feb. 2014 www.rethinkmedical.com

[20] Rizzo, G. (2013). Design and evaluation of an effective BCI-based adaptive user application: A preliminary case study, in: *User Modeling, Adaptation, and Personalization (UMAP 2013),* LNCS 7899:380-383.

[21] Royer, A.S., et al. (2010). EEG Control of a Virtual Helicopter in 3-Dimensional Space Using Intelligent Control Strategies. *IEEE Trans. on Neural Systems Rehabilitation Engineering* 18(6):581-589.

[22] Schirner, G., et al. (2013). The Future of Human-in-the-Loop Cyber-Physical Systems*, IEEE Computer* 46(1):36-45.

[23] Sellers, E., Schalk, G., Donchin, E. (2003). The P300 as a typing tool: tests of brain computer interface with an ALS patient, *Psychophysiology* 40:S77.

[24] Taylor, D.M., Tillery, S.I.H., Schwartz, A.B. (2002). Direct cortical control of 3D neuroprosthetic devices, *Science,* 296(5574):1829-1832.

[25] Velasco-Alvarez, F., et al. (2013). BCI-based navigation in virtual and real environments, *Advances in Computational Intelligence,* LNCS 7903:404-412.

[26] Velliste M, Perel S, Spalding M.C., Whitford A.S., Schwartz A.B. (2008). Cortical control of a prosthetic arm for self-feeding, *Nature* 453:1098-1101.

[27] Villegas, N.M., et al. (2011). A framework for evaluating quality-driven self-adaptive software systems, in: *Proc. 6th Int. Symp. on Software Engineering for Adaptive and Self-Managing System*s (SEAMS 2011), pp. 80-89.

[28] von Neumann, J. (2012). *The Computer and the Brain.* Yale University Press, 3rd Edition.

[29] Wearable Sensing, retrieved Feb. 2013 wearablesensing.com

[30] Whittle, J., et al. (2010). RELAX: A language to address uncertainty in self-adaptive systems requirement, in: Requirements Engineering 15(2):177-196.

[31] Wolpaw, J.R., McFarland, D.J. (2004). Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans, in: *Proc. National Academy of Sciences USA* 101(51):17849-54.