Theoretical Computer Science 623 (2016) 15–64

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Modelling and analysing neural networks using a hybrid process algebra

Robert J. Colvin

The University of Queensland, Queensland Brain Institute, Brisbane, 4072, Australia

ARTICLE INFO

Article history:
Received 13 December 2012
Received in revised form 15 August 2015
Accepted 24 August 2015
Available online 31 August 2015
Communicated by P. Degano

Keywords:
Neural networks
Process algebra
Hybrid systems

ABSTRACT

Research involving artificial neural networks has tended to be driven towards efficient computation, especially in the domain of pattern recognition, or towards elucidating biological processes in the brain. Models have become more detailed as our understanding of the biology of the brain has increased, incorporating real-time behaviour of individual neurons interacting within complex system structures and dynamics. There are few examples of abstract and fully formal models of biologically plausible neural networks: in the neural networks literature models are often presented as a mixture of mathematical equations and natural language, supported by simulation code and associated experimental results. The informality often hides or obscures important aspects of a particular model, and leaves a large conceptual gap between the model descriptions and the usually low-level programming code used to simulate them.

The main contribution of this paper is formally modelling and analysing a biologically plausible neural network model from the literature that exhibits complex neuron-level behaviour and network-level structure. To achieve this a modelling language 'Pann' is developed, based on the process algebras CSP and Hybrid$_\chi$. It is designed to be convenient for mixing the behaviour of discrete events (such as a neuron spike) with mutable continuous and discrete variables (representing chemical properties of a neuron, for instance). Its behaviour is defined using an operational semantics, from which a set of general properties of the language is proved.

The groundwork for the biological model is laid by first formalising some well-known concepts from the artificial neural networks domain, such as feedforward behaviour, backpropagation, and recurrent neural networks. The Pann model of a feedforward network, comprising a set of communicating processes representing individual neurons, is proved equivalent to the standard one-line calculation of feedforward behaviour.

1. Introduction

Artificial neural networks have been used both for their computational power and as abstractions of the behaviour of the brain for many decades. Early models exhibited a basic computational principle of a highly connected network of individual neurons that transform their input into an output and send this on to other neurons [1,2]. This model has attractive computational power, but despite emerging from brain research provides relatively little insight into the biology of real neural networks. More recent research describes the dynamics of the brain, which typically involves neurons acting in real-

E-mail address: r.colvin@uq.edu.au.

http://dx.doi.org/10.1016/j.tcs.2015.08.019
0304-3975/

time, governed by differential equations [3,4], with such models having significant computational power themselves [5–7]. Although the complexity of the neuron models and network dynamics is quite high, most descriptions of neural networks in the literature tend to mix mathematical equations with natural language, and hence may contain some ambiguity or imprecision. This may be rectified in associated simulation code, but such code is often harder to understand if written in a low-level implementation language, and in particular if it has been optimised for computational efficiency, including the need to recast continuous-time systems as discrete-time systems.

Despite the high degree of communication and concurrency between neurons and the complexity of the behaviour described, there has been relatively little work on fully formalising the description of realistic neural networks. For feedforward networks, perfectly adequate mathematical equations can describe entire systems in one line, however, learning algorithms, the behaviour of recurrent networks, and the dynamics of biologically plausible models are much less concisely described. Models of the latter category, which may include genetic, chemical, and electrophysiological factors, will become more complex and difficult to describe accurately as our knowledge of the brain increases.

The main contribution of this paper is in fully formalising and analysing a widely-used neural network that operates in real-time and has a relatively close correlation to the neurobiology of cortical neurons [8]. The formalisation (see Fig. 15) is given in a hybrid process algebra ("Pann") which is built from typical concepts in the concurrency literature ([9–11] and especially Hybrid$_\chi$ [12]). Its behaviour is defined using an operational semantics [13–15], using a novel approach for hybrid concepts that builds on earlier work for discrete systems [16,17]. The semantics is used to derive a set of equations for reasoning about processes, which are used to elucidate properties of the model. In addition, we formalise some concepts from the artificial neural network literature (see, e.g., Fig. 6), and derive an apparently novel result that formally equates the standard mathematical description of feedforward neural networks to a highly parallel description that mixes local variables and interprocess communication.

The emphasis in this paper is in providing compact models for the description of the systems found in this paper rather than a more general notation, although it is the intention that extensions to Pann and its semantics can be accommodated. The meaning of a Pann process is formed from the sequences of atomic steps (traces) it may take, which involve communication with other processes and changes to local and shared variables, including changes in real-time. The main difference between the semantics given here and other approaches lies in the handling of variable declarations, which we argue is simpler and more abstract, and convenient for describing systems such as neural networks that contain many parallel processes (neurons) which have their own local state and also interact via shared state.

The motivation of the work is to open the rapidly developing field of neural network modelling to formal analysis by methods from theoretical computer science. Describing neural networks in a uniform and fully formal way has potential application because there is an increasing need to integrate the vast amount of neurological data that is being collected across a range of tools and techniques, which cover measurements on the scale of individual cells to anatomical regions. Further complicating neuroscientific research in humans is that underlying biological functions must eventually be linked to complex cognitive functions. Concepts and techniques from theoretical computer science for abstraction and analysis may therefore be of benefit. However it would be unrealistic to assume that neural network researchers will find immediate benefit in this paper, as understanding process algebras and their semantics is a non-trivial undertaking itself; but, as argued by Fisher and Henzinger [18], operational descriptions of biological systems provide the foundation for collaboration between modellers and biologists, and it is possible that a biologist with little exposure to either formal modelling languages or mathematical/simulation based modelling may find the former easier to interrogate when working collaboratively.

Section 2 gives an introduction to the classic feedforward neural network and backpropagation algorithm in terms of standard mathematical equations. Section 3 introduces the discrete variable/event subset of Pann and uses it to recast the mathematical neural network models of Section 2, as well as recurrent neural network behaviour. Section 4 formally describes a model of a neural network that includes real-time behaviour, differential equations to describe changes in variables over time, and complex temporal relationships between neurons. For these purposes we use a model described by Izhikevich [8], due to its relatively simple (and computationally relatively efficient) local dynamics, but more interesting temporal dynamics and structure (more complex equations, as in the Hodgkin–Huxley model [19], can be substituted without affecting the underlying language or its semantics). Section 5 gives the formal definition of the discrete subset of Pann, which is extended in Section 6 to include real-time and hybrid concepts. Related work is discussed in Section 7 and conclusions in Section 8. Appendix A contains the proof that the process-oriented description of a feedforward network is equivalent to its mathematical description, and Appendix B explores the consequences of using (CCS-like) binary synchronisation instead of (CSP-like) multi-way synchronisation as the communication mechanism.

2. State-based formalisation of a feedforward neural network with backpropagation

In this section we give a specification of a typical feedforward neural network with backpropagation, serving as an introduction to artificial neural networks and formal languages. The description of feedforward and backpropagation behaviour are derived mainly from Bishop [20][1]; see also, e.g., [21].

[1] To avoid distraction the networks do not include bias units, although their addition is straightforward.