Microelectronics Journal 45 (2014) 690–701

Contents lists available at ScienceDirect

Microelectronics Journal

journal homepage: www.elsevier.com/locate/mejo

An FPGA based scalable architecture of a stochastic state point process filter (SSPPF) to track the nonlinear dynamics underlying neural spiking

Yao Xin [a], Will X.Y. Li [a,*], Ray C.C. Cheung [a], Rosa H.M. Chan [a], Hong Yan [a], Dong Song [b], Theodore W. Berger [b]

[a] Department of Electronic Engineering, City University of Hong Kong, Hong Kong SAR
[b] Center for Neural Engineering, Department of Biomedical Engineering, University of Southern California, Los Angeles, CA 90089, USA

ARTICLE INFO

Article history:
Received 23 August 2013
Received in revised form
27 March 2014
Accepted 31 March 2014
Available online 10 May 2014

Keywords:
SSPPF
Adaptive filter
Neural modeling
Field-programmable gate array

ABSTRACT

Recent studies have verified the efficiency of stochastic state point process filter (SSPPF) in coefficients tracking in the modeling of the mammalian nervous system. In this study, a hardware architecture of SSPPF is both designed and implemented on a field-programmable gate array (FPGA). It provides a time-efficient method to investigate the nonlinear neural dynamics through coefficients tracking of a generalized Laguerre–Volterra model describing the spike train transformations of different brain sub-regions. The proposed architecture is able to process matrices and vectors with arbitrary sizes. It is designed to be scalable in parallel degree and to provide different customizable levels of parallelism, by exploring the intrinsic parallelism of the FPGA. Multiple architectures with different degrees of parallelism are explored. This design maintains numerical precision and the proposed parallel architectures for coefficients estimation are also much more power efficient.

1. Introduction

Cognitive neural prosthesis design is an emerging topic in neural engineering research. For many years, various studies have endeavored to develop a silicon-based prosthetic device that can be implanted into the mammalian brain, e.g., the hippocampal CA3 [1]. This device is expected to perform bi-directional communications between the intact brain regions while bypassing the degenerated CA3 region. Such a prosthetic device (if successfully developed) could provide fundamental treatment to diseases related to cognitive impairment, such as Alzheimers.

The generalized Laguerre–Volterra model (GLVM) proposed by Song et al. [2] is a data-driven model first applied to predict mammalian hippocampal CA1 neuronal spiking activity based on detected CA3 spike trains by which the expected neuroprosthetic function can be achieved. GLVM consists of five major components (Fig. 1). It uses a weighted sum of convolution products between the model inputs and the orthonormal Laguerre basis functions, passing through a threshold trigger to generate the predicted model outputs. A detailed calculation flow of the generalized Laguerre–Volterra (GLV) algorithm can be found in [2,3]. The prediction function of the GLV algorithm is very straightforward when implemented on different platforms. The Laguerre coefficients (weights of convolution products), however, must be estimated first using the recorded input/output data. This estimation process is often the most computationally intensive stage in the entire calculation flow. In this work, we focus on the estimation stage of the GLV algorithm and utilize an efficient adaptive method to estimate the coefficients dynamically and accurately.

The stochastic state point process filter (SSPPF) is a suitable choice for the aforementioned problem [4]. Derived by Eden et al. [4] based on Bayes' rule Chapman–Kolmogorov paradigm and point process observation models, it has been verified to be effective for tracking dynamics of neural receptive fields under various conditions. In 2009, Chan et al. [5] applied the algorithm to realize the estimation function of the GLVM.

GLVM is only one successful application of SSPPF which is designed for point process and especially suitable to do neural encoding/decoding. This filter has been compared with SDPPF, extended Kalman filter (EKF), and pass-by-pass method [4] in the simulation of the response of pyramidal neurons in the CA1 region of the rat hippocampus to the movement of the animal. SSPPF

* Corresponding author.
E-mail addresses: yaoxin2-c@my.cityu.edu.hk (Y. Xin), xyli@ee.cityu.edu.hk (W.X.Y. Li), r.cheung@cityu.edu.hk (R.C.C. Cheung), rosachan@cityu.edu.hk (R.H.M. Chan), h.yan@cityu.edu.hk (H. Yan), dsong@usc.edu (D. Song), berger@bmsr.usc.edu (T.W. Berger).

http://dx.doi.org/10.1016/j.mejo.2014.03.018
0026-2692/

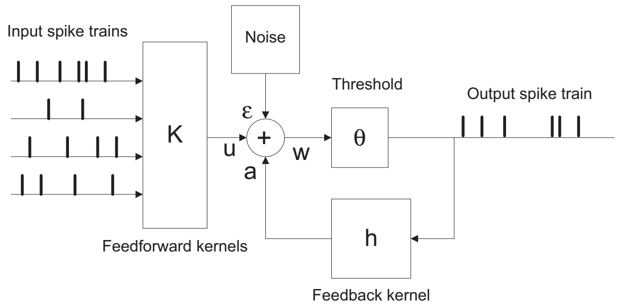

Fig. 1. Major components of the generalized Laguerre–Volterra model: K is the feedforward Volterra kernels, h is the feedback kernel, θ is a threshold quantity, ϵ is the Gaussian white noise quantity. The u, a and w are synaptic potential, after potential and pre-synaptic membrane potential respectively. For details see Section 2 of [3].

provided the most accurate tracking of both the linear (slow) and jump (rapid) evolution scenarios. In the adaptive decoding study [4], the feasibility of reconstructing a neural signal with SSPPF from an ensemble whose receptive fields are evolving is illustrated. Moreover, Salimpour et al. [6] have applied the SSPPF to the neural spiking activity of inferior temporal cortex of macaque monkey for the first time. The filter is used to estimate the conditional intensity of the point process observation as a time varying firing rate in cooperation with a particle filter. The results with a real data indicate that, based on the assessment of goodness-of-fit, the neural spiking activity and the biophysical property of neuron could be captured more accurately compared with conventional methods.

Although the SSPPF has been successfully and widely applied, due to the fact that this algorithm was implemented with commercial software and run on a desktop setup, the calculation process has certain limitations. Although conventional CPUs have experienced a significant improvement on computing power, there still exist many bottlenecks, such as the difficulty to further increase clock rates and a mismatch between memory bandwidth and ever-increasing CPU speed, which are preventing them from successfully keeping up with the demands from high-performance computing (HPC) applications. When the number of GLVM input and output grows, the number of Laguerre coefficients to estimate increases exponentially. Therefore, a high-performance platform is required for off-line (for model training) and on-line (for prediction) coefficient estimations.

In order to fill the enlarged gap between HPC requirement and limited CPU performance, other parallel platforms are attracting attentions from researchers. The FPGA is a competitive solution which is reconfigurable to satisfy various computational requirements, like large-scale data or real-time video processing. It is able to provide different levels of parallelism according to available hardware resource to maximize the performance. More importantly, the inherent low-power property of FPGAs makes it suitable for implantable neural prosthesis designs. Furthermore, the FPGA design is always demanded as a prototype of silicon-based prosthetic device for verification and testing.

In this work, we overcome the limitations of the previous works and for the first time, implement the SSPPF on the FPGA-based hardware platform to achieve more efficient and effective model coefficients estimation.

The major contributions of this paper are as follows: (1) The first hardware architecture of SSPPF has been designed and implemented, which is practical to be applied to the general framework of future cognitive prosthetic device. (2) This design is capable of processing arbitrary size matrices/vectors without any pre-configurations; the maximal size supported is limited only by the available on-chip memory resource. (3) In order to achieve high performance computing, the architecture is made scalable through parallelism. Multiple designs in different parallelism degrees are explored, implemented, and tested. (4) Our design can be generalized to readily accommodate other adaptive filters which require extensive matrix/vector operations, since the essential matrix operations (multiplication and inversion) are realized in current work.

The rest of the paper is organized as follows: Section 2 describes related studies. Section 3 introduces the SSPPF algorithm. Section 4 describes both the overall hardware architecture and its components. Section 5 provides the implementation results and performance evaluation. Section 6 summarizes the paper.

2. Related works

The performance of a platform established for conducting the estimation function can be measured with three standards: effectiveness, efficiency, and realisticity of application. A silicon-based implementation of the GLVM dates back to 2005, when Berger et al. [7] first completed initial prototyping work utilizing a field-programmable gate array (FPGA). In 2006, Hsiao et al. [8] fabricated such an architecture using the 18 μm process of Application Specific Integrated Circuit (ASIC) technology. Their study was based on the single-input, single-output (SISO) GLVM, the simplest form of the model. However, in a real-world situation, model output is often affected by the spiking activity of multiple inputs. This activity weakens the realisticity of the design application into a practical neuroprosthetic device.

In 2011, Li et al. [9] successfully implemented the multi-input, multi-output (MIMO) GLVM on the FPGA-based hardware platform. They achieved a remarkable speedup in model coefficient estimation when compared to a traditional software based approach. They adopted the steepest decent point process filter (SDPPF) as the tracking method. This method is simpler in the mathematical representation but sacrifices certain levels of accuracy when compared to other well-established methods, i.e., the SSPPF or the Kalman Filter. Thus, this method is less effective.

A major improvement of SSPPF over SDPPF is the introduction of the adaptive learning rate. Learning rate is important for

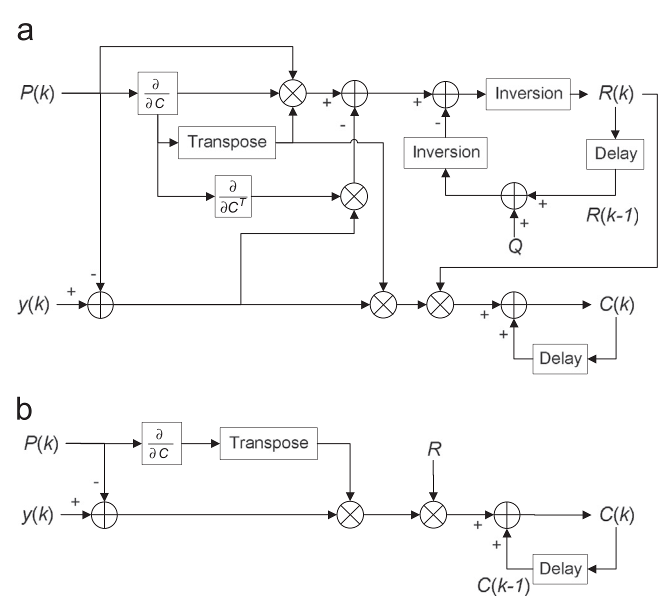

Fig. 2. Diagrams of SSPPF and SDPPF flow.