LOIHI DEEP DIVE

Architecture, SDK, Examples

Neuromorphic Computing Lab | Intel Labs

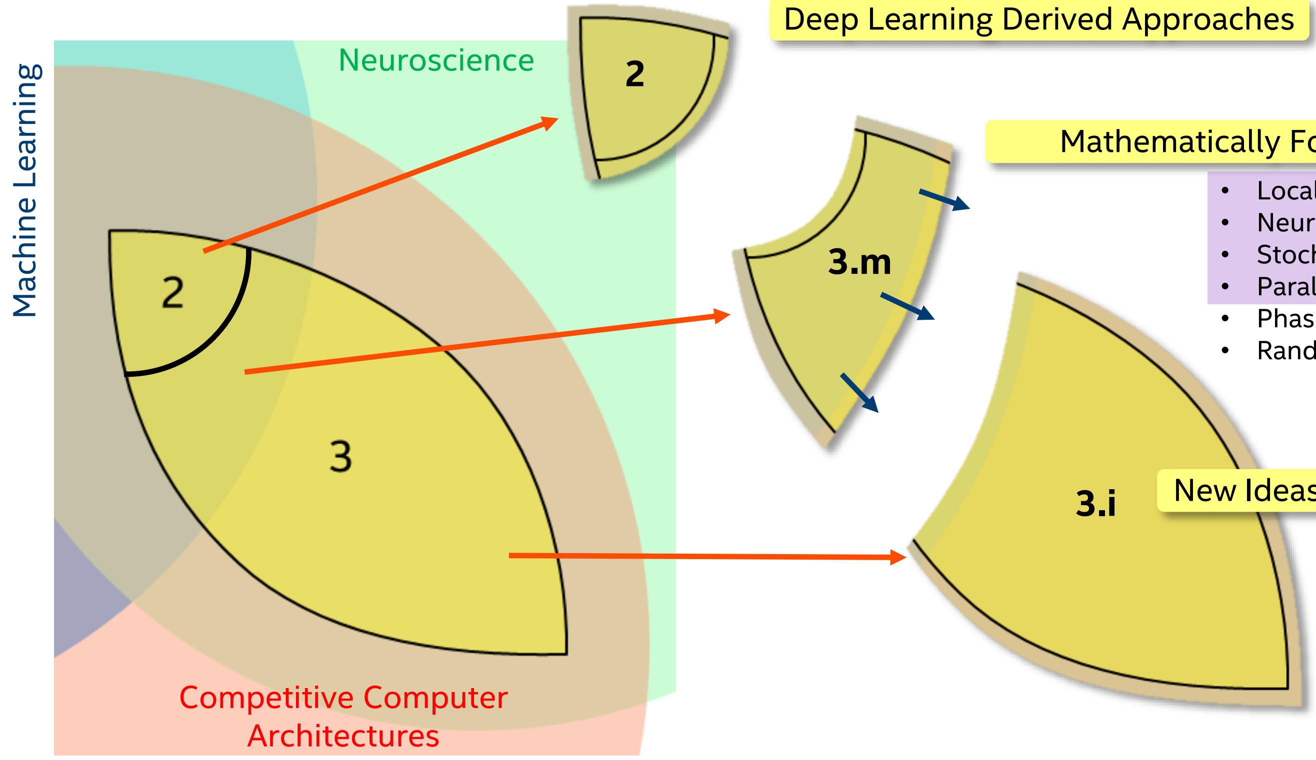
March 29, 2019
Neuro-Inspired Computational Elements, SUNY Polytechnic Institute

SCHEDULE FOR THE DAY

Morning Session	
9:15 – 10:15	Loihi Architecture Overview
10:15 – 10:25	Break / Hardware Q&A
10:25 – 10:55	NxSDK Architecture
11:00 – 11:35	NxNet Intro • Add compartments/connections (code, basic behavior) • STDP Learning and eligibility traces • Kapoho Bay DVS Demo • Multi-compartment neurons – *time permitting*
11:35 – 11:45	Software Q&A

Afternoon Session	
1:00 – 1:10	NxSDK Overview (quick version)
1:10 – 2:05	Algorithmic Demos with NxNet • Single-Layer Image Classification • Solving LASSO w/ Spiking LCA • Constraint satisfaction
2:05 – 2:35	Algorithmic Demos with Nengo • Nonlinear oscillator • Learning w/ Prescribed Error Sensitivity • MNIST classification with Nengo DL • Keyword spotting with Nengo DL
2:35 – 2:50	Graph Search and Multi-Chip Scaling
2:50 – 3:00	Closing / Q&A

SNN Algorithms Discovery and Development



Deep Learning Derived Approaches

- DNN -> SNN conversion
- SNN backpropagation
- Online SNN pseudo-backprop

Mathematically Formalized

- Locally Competitive Algorithm for LASSO
- Neural Engineering Framework (NEF)
- Stochastic SNNs for solving CSPs
- Parallel graph search
- Phasor associative memories
- Random diffusion walkers

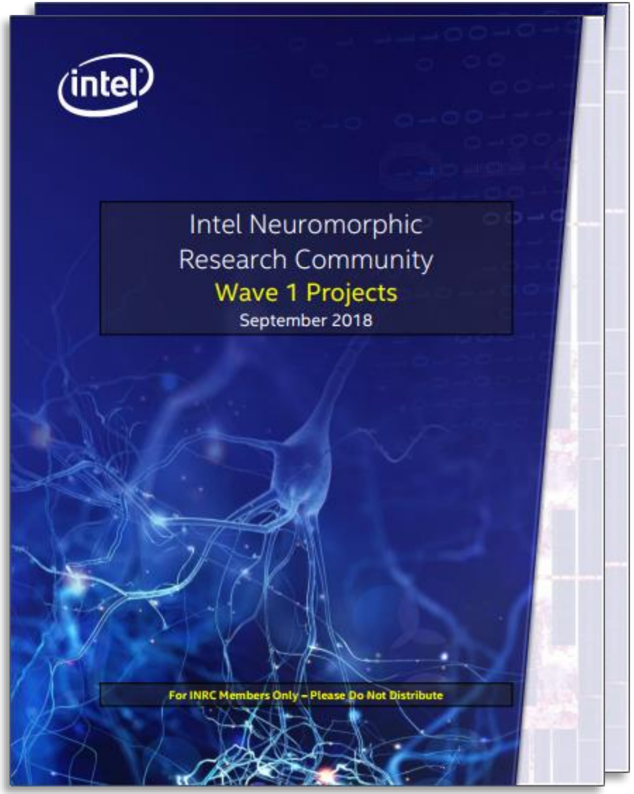
New Ideas Guided by Neuroscience

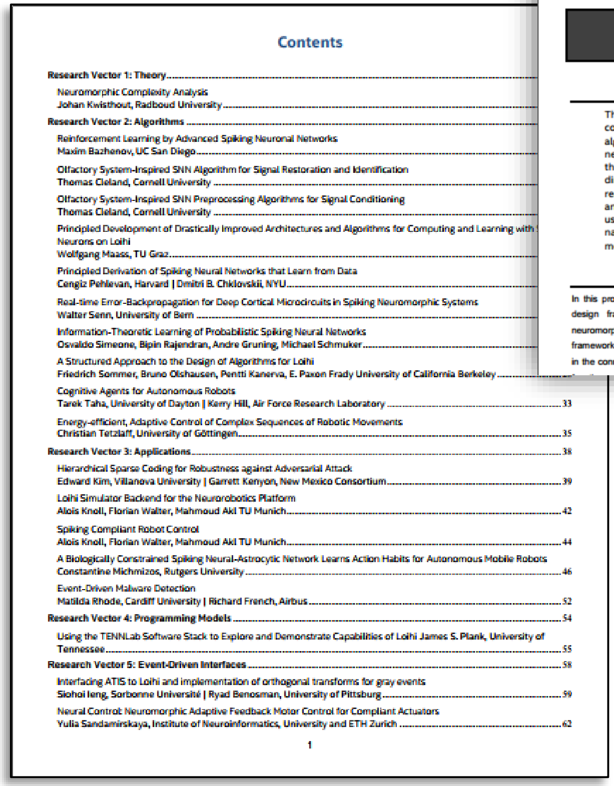
- Olfaction-inspired rapid learning
- Dynamic Neural Fields
- SLAM
- Evolutionary search
- Cortical models

Intel's Objectives for INRC

1) **Accelerate** research in neuromorphic computing
   By stimulating *algorithms* and *applications* research focusing on Loihi architecture
2) **Quantify** the value of neuromorphic computing today
   Discipline of a quantified approach is critical for progress and mainstream adoption
3) **Inform Loihi's architectural development**
   Algorithms & application findings provide insights for future silicon revisions
4) **Build an ecosystem** that can provide a market for neuromorphic chips
   Intel hopes to sell chips, eventually… we need customers and a broad user base

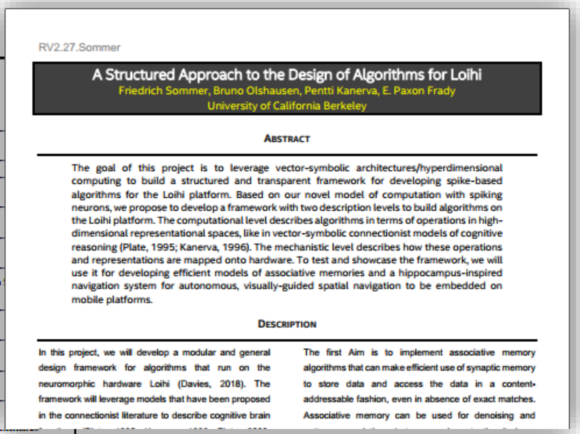
Join the Community





Contents

1



RV2.27.Sommer

A Structured Approach to the Design of Algorithms for Loihi

Friedrich Sommer, Bruno Olshausen, Pentti Kanerva, E. Paxon Frady
University of California Berkeley

ABSTRACT

The goal of this project is to leverage vector-symbolic architectures/hyperdimensional computing to build a structured and transparent framework for developing spike-based algorithms for the Loihi platform. Based on our novel model of computation with spiking neurons, we propose to develop a framework with two description levels to build algorithms on the Loihi platform. The computational level describes algorithms in terms of operations in high-dimensional representational spaces, like in vector-symbolic connectionist models of cognitive reasoning (Plate, 1995; Kanerva, 1996). The mechanistic level describes how these operations and representations are mapped onto hardware. To test and showcase the framework, we will use it for developing efficient models of associative memories and a hippocampus-inspired navigation system for autonomous, visually-guided spatial navigation to be embedded on mobile platforms.

DESCRIPTION

In this project, we will develop a modular and general design framework for algorithms that run on the neuromorphic hardware Loihi (Davies, 2018). The framework will leverage models that have been proposed in the connectionist literature to describe cognitive brain

The first Aim is to implement associative memory algorithms that can make efficient use of synaptic memory to store data and access the data in a content-addressable fashion, even in absence of exact matches. Associative memory can be used for denoising and

Available to members:

- Access to member website
- Project documentation
- Access to GitHub site
- Participation in INRC workshops

INRC Engagement Process

1) Email **inrc_interest@intel.com**
   We'll send you our RFP and project proposal template
2) Submit a **project proposal**
   Tell us what you want to investigate and accomplish with Loihi
3) Execute the **INRC participation agreement**
   Requires signature of someone who can legally bind your organization
4) Receive **Neuromorphic Research Cloud** accounts
   You get a private VM on our system + accounts for your team members
5) Request **Loihi hardware**
   We'll loan you physical systems when and if you need them…

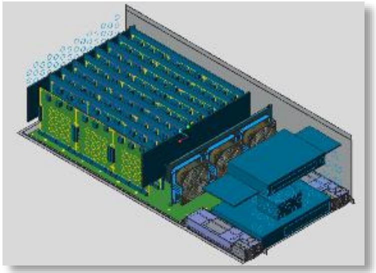
Loihi Systems

Q4 2017
Wolf Mountain
Remote Access
4 Loihi/Board

Q2 2018
Nahuku
Arria10 Expansion Board
For cloud & local use
8-32 Loihi/Board

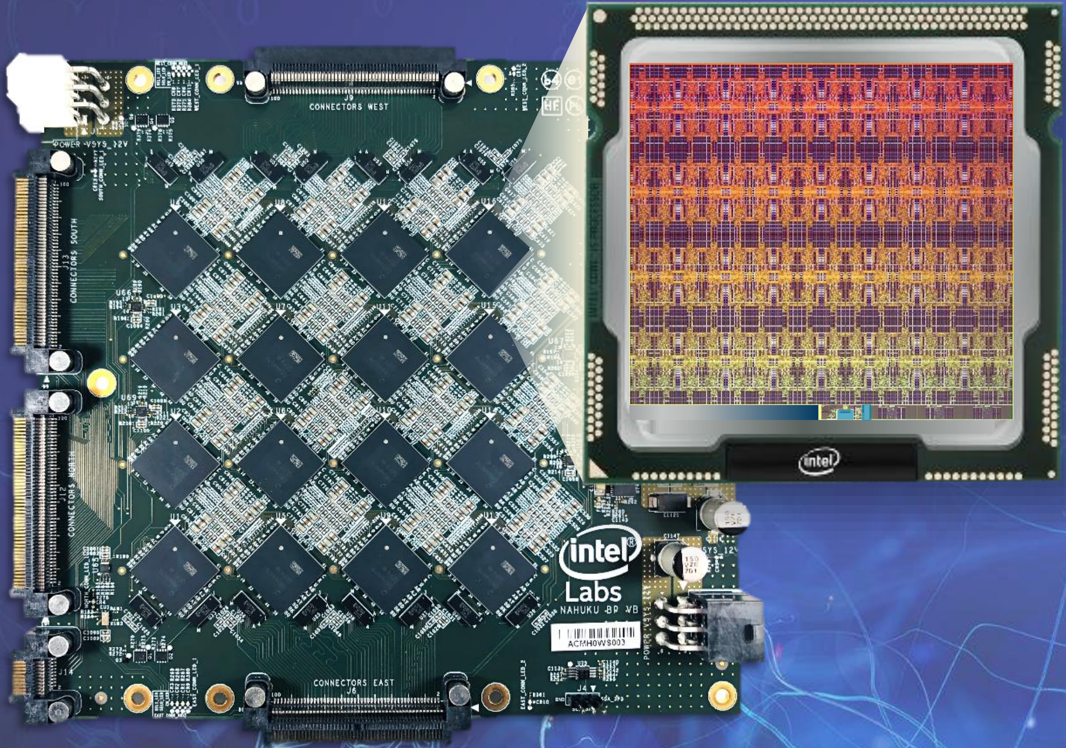
Q3 2018
Kapoho Bay
1-2 Loihi
DVS interface
USB host interface

Q2 2019
Pohoiki Springs
Remote Access
Up to 768 chips
(100M neurons)

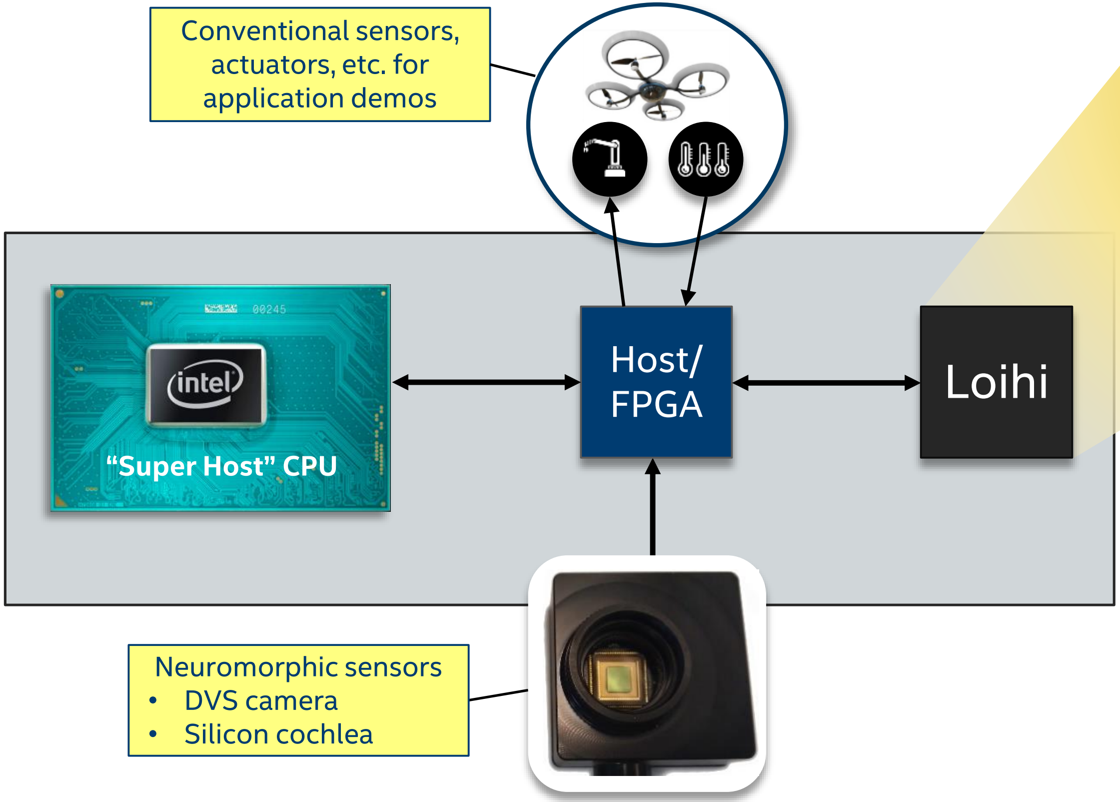


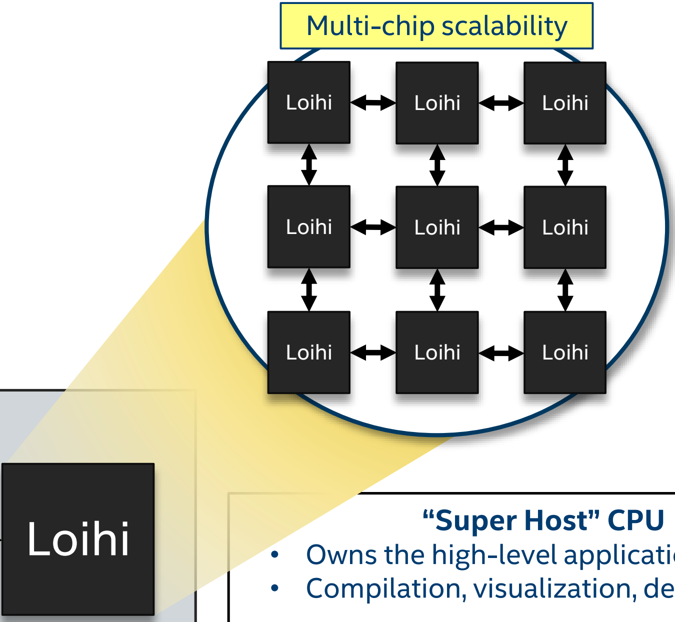
NAHUKU 32-CHIP PLATFORM

- 32 Loihi chips
- 4 x 4 mesh of chips
- Top and bottom sides



System Architecture





"Super Host" CPU
- Owns the high-level application
- Compilation, visualization, debug, UI

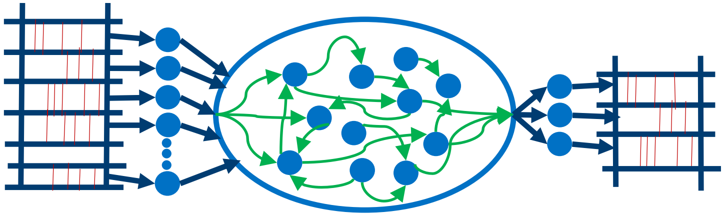
FPGA/Host
- Manages Loihi chips
- Interfaces to outside world

Loihi x86
- System management
- Some custom user code

Loihi Neurocore
- Spiking neural network hardware

SDK Architectural Considerations

Abstract Network Definition



Objective

efficiently map abstract spiking neural network definitions onto our heterogeneous hierarchical implementation

Architectural Principles

Programmability – Must be accessible in the lingua franca of machine learning (python) and at multiple levels of abstraction.

Simplicity and Modularity– Hardware details are abstracted away. Easy things are easy and complexity is added incrementally. Functionality available through modular interfaces.

Efficiency and Scalability – Additional hardware resource retains efficiency and adds to performance and scalability not to complexity.

Observability – Rich ability to probe and monitor networks as they execute.

Flexibility/Extensibility– Designed to snap into higher level APIs and enable a variety of sensors and actuators.

Software Development Kit

Heterogeneous, Hierarchical Hardware

Loihi System

Nx SDK Software Architecture

A module is a complete NxNet defined algorithm (I/O, documentation, etc.)

Computational Modules

LCA | LSNN | EPL | VSA | TPAM

SLIC | CSS | Path Planning | DNF | Astro

3rd party APIs and Frameworks

Nengo | EONS | NRP

ROS | Tensorflow | PyNN

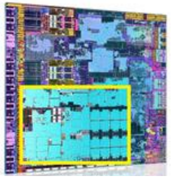
NxNet API

Sequential Neural Interfacing Processes (SNIPs)

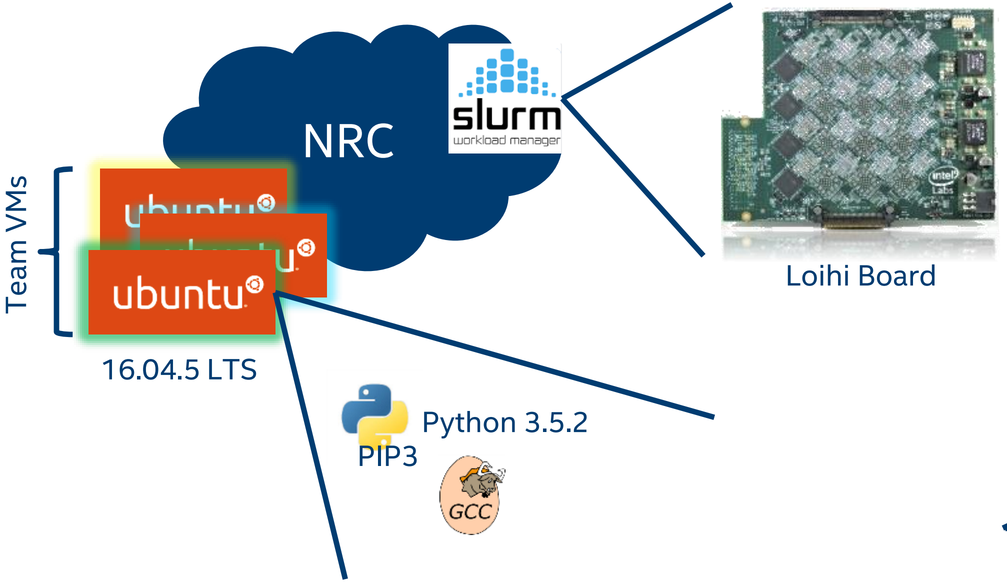
Spiking Neural Network (SNN)

NxCompiler

NxCore/NxDriver/NxRuntime



Toolchain



sinfo – see what's available
squeue – see what's running
scancel – stop your job if it gets out of hand

* Other names and brands may be claimed as the property of others

Thank You!

Email inrc_interest@intel.com for more information

LEGAL INFORMATION