

INTRODUCTION TO THE IPU

GRAPHCORE



Neural network visualization from [POPLAR™](#)

The Hardware Lottery

Sara Hooker

Google Research, Brain Team

shooker@google.com

Abstract

Hardware, systems and algorithms research communities have historically had different incentive structures and fluctuating motivation to engage with each other explicitly. This historical treatment is odd given that hardware and software have frequently determined which research ideas succeed (and fail). This essay introduces the term hardware lottery to describe when a research idea wins because it is suited to the available software and hardware and *not* because the idea is superior to alternative research directions.

Examples from early computer science history illustrate how hardware lotteries can delay research progress by casting successful ideas as failures. These lessons are particularly salient given the advent of domain specialized hardware which make it increasingly costly to stray off of the beaten path of research ideas. This essay posits that the gains from progress in computing are likely to become even more uneven, with certain research directions moving into the fast-lane while progress on others is further obstructed.

OUR IPU LETS INNOVATORS CREATE THE NEXT BREAKTHROUGHS IN MACHINE INTELLIGENCE



GOOGLE'S AI GURU WANTS COMPUTERS TO THINK MORE LIKE BRAINS



WIRED

Wired – “How might we build machine learning systems that function more like a brain? ”

Geoff Hinton – “I think we need to move towards a different type of computer. Fortunately I have one here...”
Hinton reaches into his wallet and pulls out a large, shiny silicon chip:



an IPU processor from Graphcore

A DELIBERATELY DIFFERENT TECHNOLOGY

- 2 key differentiators: memory access speed and parallelism
- Orders of magnitude more **parallelism** due to **MIMD** architecture (multiple instruction, multiple data): thousands of independent, programmable cores
- 30x the **memory bandwidth** of GPU due to enormous amount of **memory on the chip**
- Disaggregated approach that allows users to configure **AI compute to CPU** ratio to optimise performance while also maximising power and space available in data centre



PARALLELISM & MEMORY ACCESS

CPU

GPU

IPU

Parallelism

Suitable for scalar processes

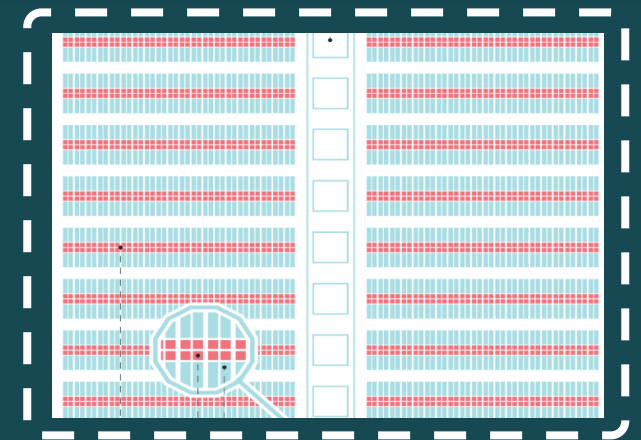
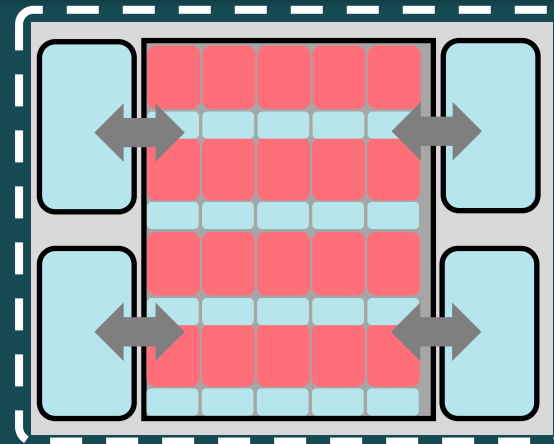
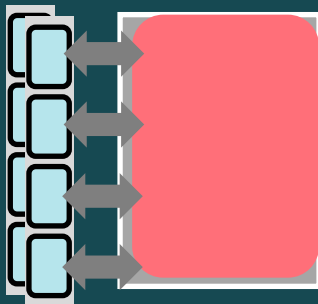
SIMD/SIMT architecture. Suitable for large blocks of dense contiguous data

Massively parallel MIMD. High performance/efficiency as ML trends to sparsity & small kernels

Processor



Memory



Memory Access

Off-chip memory

Model and Data spread across off-chip and small on-chip cache and shared mem.

Model & Data in tightly coupled large locally distributed SRAM



COMPUTE

THE WORLD'S MOST COMPLEX PROCESSOR

COLOSSUS MK2 IPU

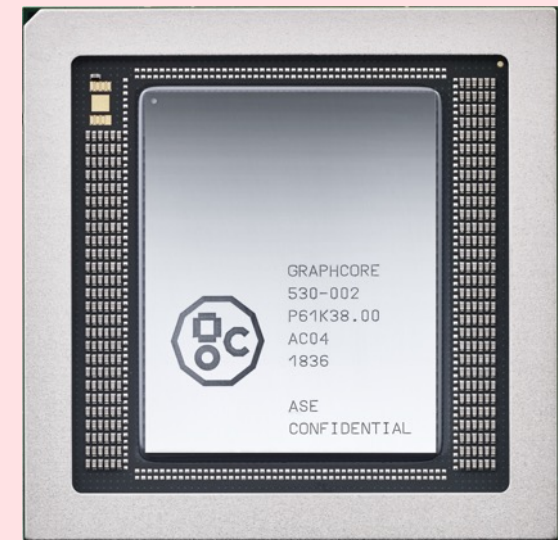
59.4Bn transistors, TSMC 7nm @ 823mm²

250TFlops AI-Float | 900MB In-Processor-Memory™

1472 independent processor cores

8832 separate parallel threads

>8x step-up in system performance vs Mk1



GC200 IPU



IPU-Tiles™

1472 independent IPU-Tiles™ each with an IPU-Core™ and In-Processor-Memory™

IPU-Core™

1472 independent IPU-Core™

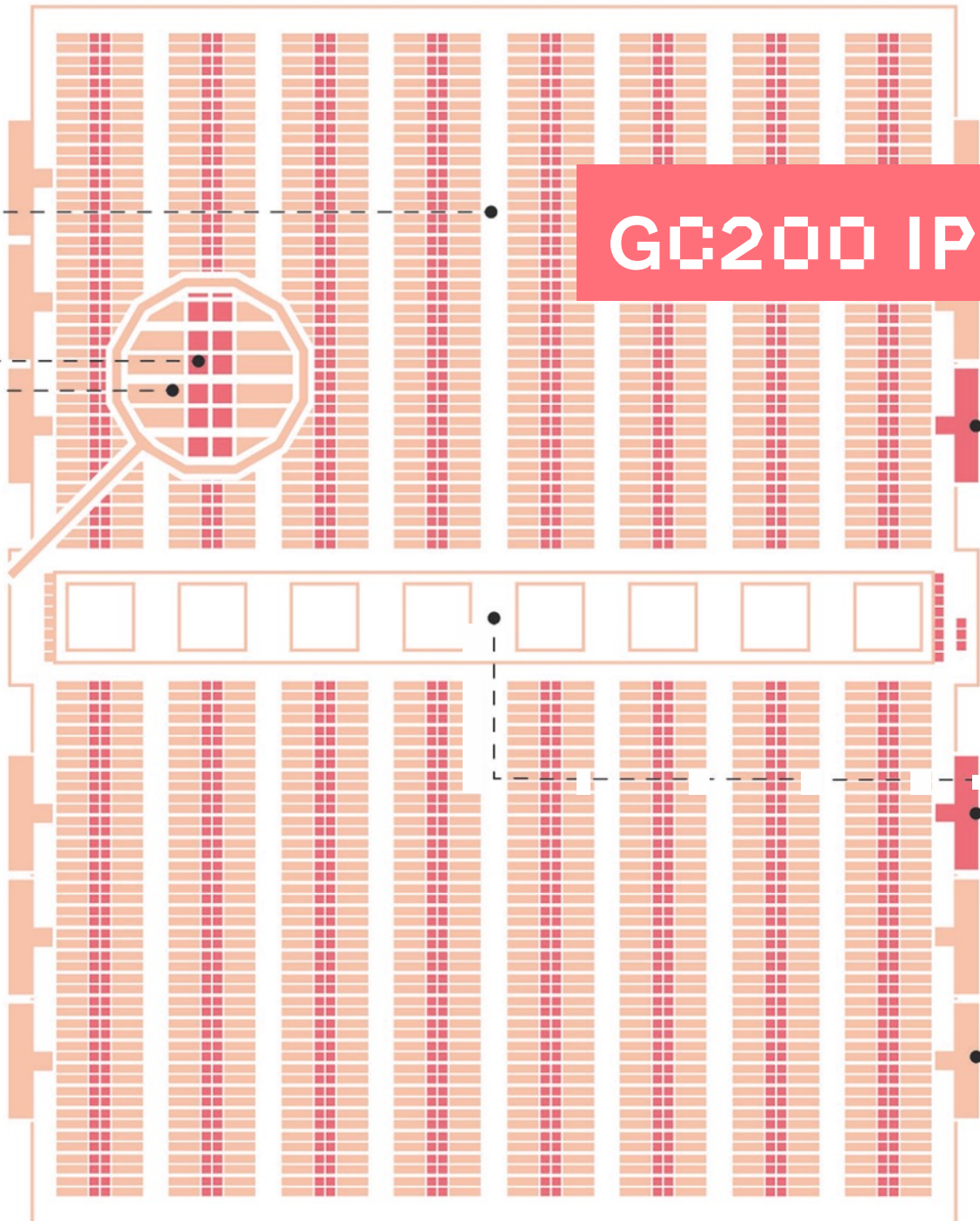
8832 independent program threads executing in parallel

In-Processor-Memory™

900MB In-Processor-Memory™ per IPU

47.5TB/s memory bandwidth per IPU

GC200 IPU PROCESSOR



IPU-Exchange™

8 TB/s all to all IPU-Exchange™
Non-blocking, any communication pattern

PCIe

PCI Gen4 x16
64 GB/s bidirectional bandwidth to host

IPU-Links™

10 x IPU-Links,
320GB/s chip to chip bandwidth



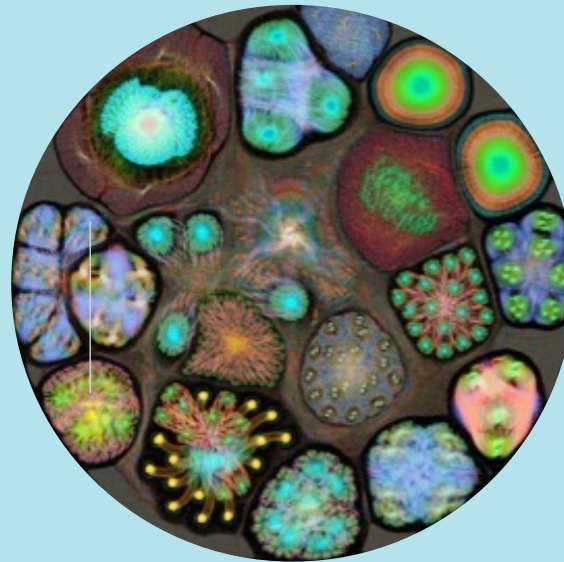
NEXT GEN AI SOLUTIONS

Hardware



IPU processors designed for AI

Software

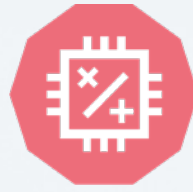


Poplar® software stack & development tools

Platforms



M2000 and Server IPU-POD₆₄ scale-out



COMPUTE



DATA

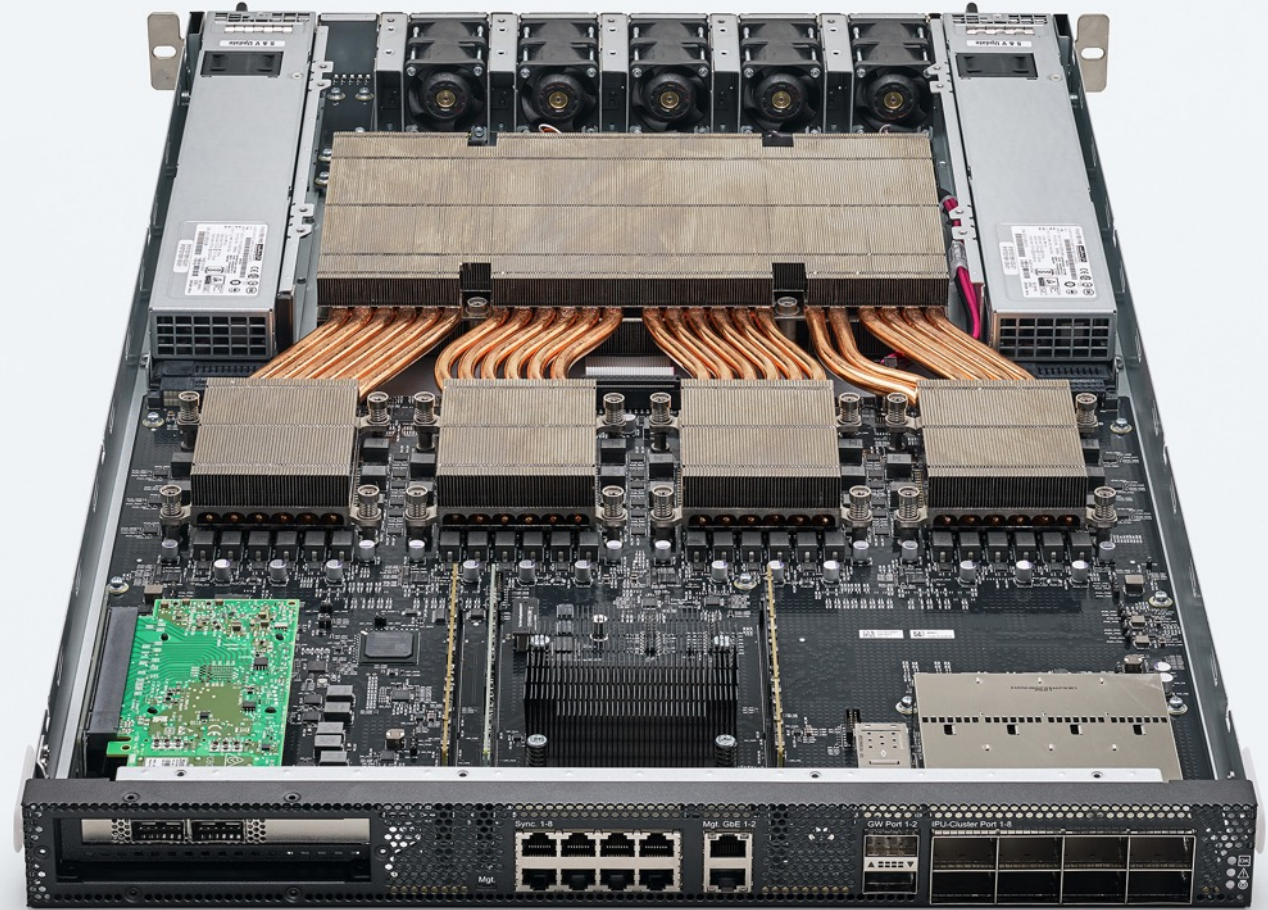


COMMUNICATIONS

IPU-MACHINE

IPU-M2000

1 PetaFlop IPU compute
2.8Tbps IPU-Fabric™



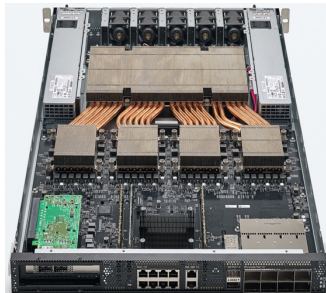
MAXIMISING ROI ON INNOVATION

M2000 – a building block for next gen data centers



GC200

x4



IPU-POD₄

4 IPU
1 PetaFlop

x4



IPU-POD16 Direct Attach

IPU-POD₁₆

16 IPU
4 PetaFlop
4x IPU-M2000

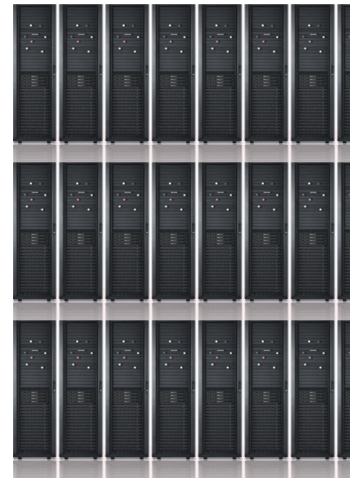
x4



IPU-POD₆₄

64 IPU
16 PetaFlop
16x IPU-M2000

x1024



IPU-POD_{64k}

64k IPU
16 ExaFlop
<7.4 PB

THE IPU LETS INNOVATORS CREATE THE NEXT GENERATION OF MACHINE INTELLIGENCE

WORKLOAD EXAMPLES GUIDE

Compute intensive benefits from significant amount of compute

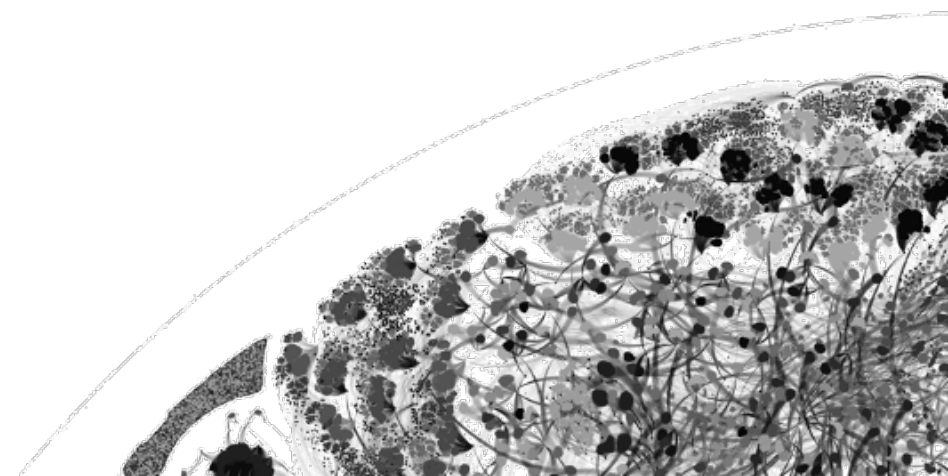
Sparse or fine-grain leverages the unique architecture

Sequential leverages the fast in-processor memory

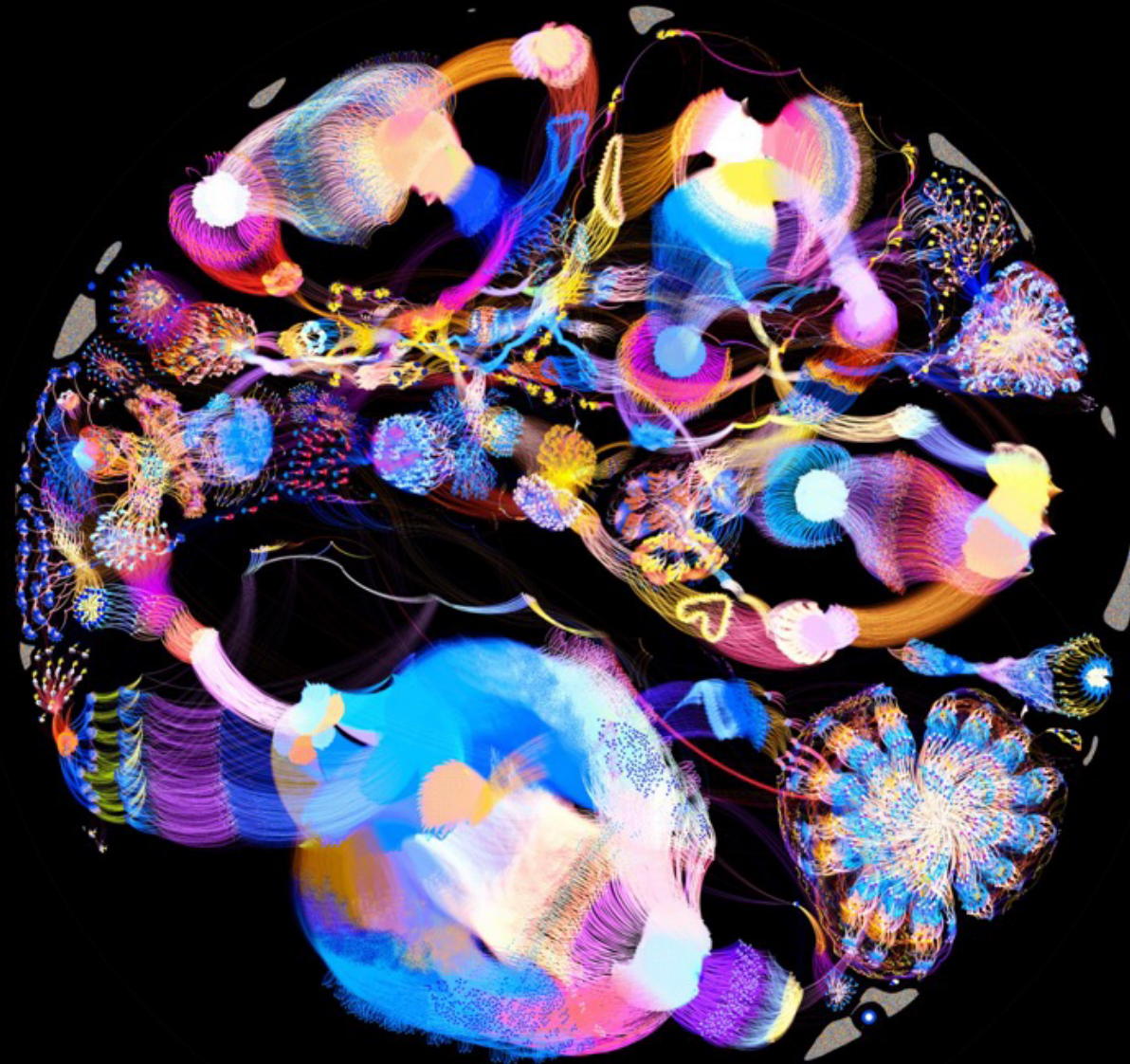
Nontrivial leverages the available parallelism

Hard to vectorise but possible to parallelise

Not bottlenecked by host compute or external I/O



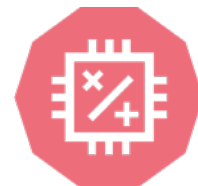
SOFTWARE



POPLAR™ COMPUTE GRAPH VISUALISATION



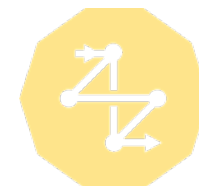
POPLAR-SDK



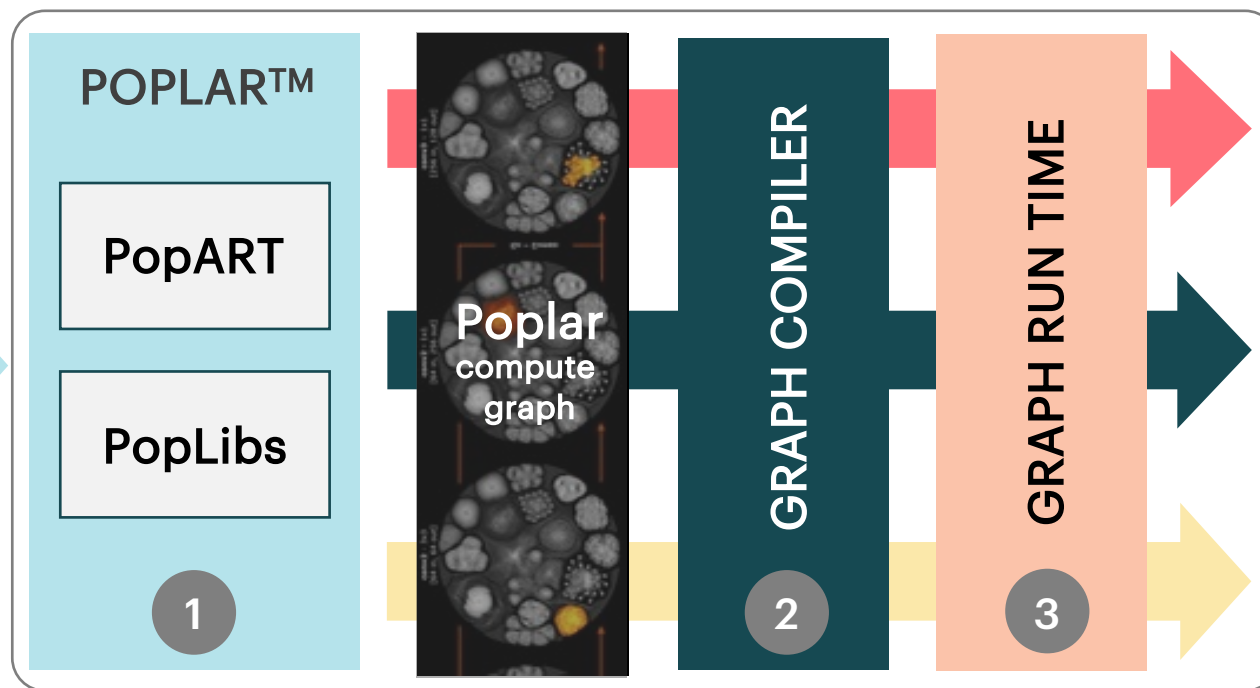
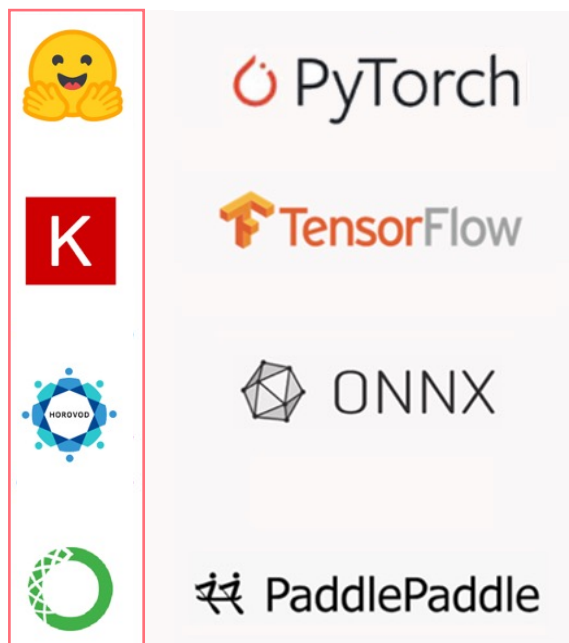
COMPUTE



DATA



COMMUNICATIONS



1,000's of IPUs
+ compiled communications



POPLAR EASE OF USE



Open & Extensible Poplar Libraries

Get access to 50+ optimised functions for common ML models and 750 high performance compute elements. Modify and write custom libraries.

ML Frameworks Support

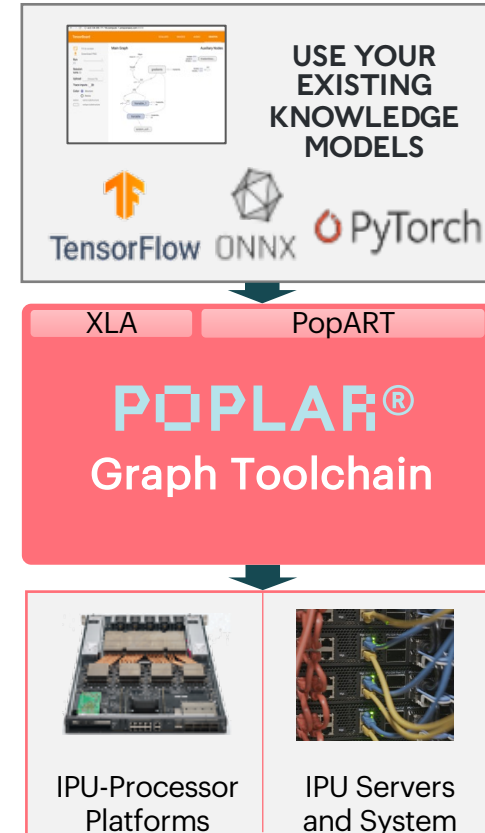
Support for standard ML frameworks: TensorFlow 1 and 2, ONNX and PyTorch with PaddlePaddle coming soon.

Straightforward Deployment

Pre-built Docker containers with Poplar SDK, Tools and frameworks images to get up and running fast.

Standard Ecosystem Support

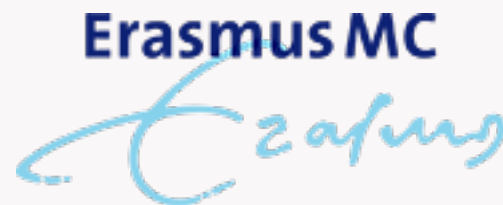
Ready for production with Microsoft Azure deployment, Kubernetes orchestration, Docker containers and Hyper-V virtualization & security.



IPI Performance



ACADEMIC & RESEARCH ENGAGEMENTS



IPU ACCELERATION FOR COSMOLOGY APPLICATIONS



Université
de Paris

Comparison of Graphcore IPUs and Nvidia GPUs for cosmology applications

Bastien Arcelin^{1*}

¹Université de Paris, CNRS, Astroparticule et Cosmologie, F-75013
Paris, France

Abstract. This paper represents the first investigation of the suitability and performance of Graphcore Intelligence Processing Units (IPUs) for deep learning applications in cosmology. It presents the benchmark between a Nvidia V100 GPU and a Graphcore MK1 (GC2) IPU on three cosmological use cases: a classical deep neural network and a Bayesian neural network (BNN) for galaxy shape estimation, and a generative network for galaxy images simulation. The results suggest that IPUs could be a potential avenue to address the increasing computation needs in cosmology.

Contents

1	Introduction	2
2	Hardware description	3
3	Cosmological use cases	4
3.1	Training data	4
3.2	Galaxy shape parameter estimation	5
3.2.1	Deterministic neural network	5
3.2.2	Bayesian neural network	6
3.3	Galaxy image generation	7
3.3.1	Generative model	8
3.3.2	Results	8
4	Summary and discussion	9
5	Acknowledgements	9

*E-mail: arcelin@apc.in2p3.fr (APC)

arXiv:2106.02465v1 [physics.comp-ph] 4 Jun 2021

Jul 09, 2021 | Research, University

UNIVERSITÉ DE PARIS ACCELERATES COSMOLOGY APPLICATIONS WITH GRAPHCORE IPUS

Written By:

Alex Titterton



This paper represents the first investigation of the suitability and performance of Graphcore Intelligence Processing Units (IPUs) for deep learning applications in cosmology””on three cosmological use cases: a classical deep neural network and a Bayesian neural network (BNN) for galaxy shape estimation, and a generative network for galaxy images production.”

The results show that **IPUs can accelerate various cosmology applications, outperforming GPUs in some cases by as much as 4x faster time to train**”



<https://www.graphcore.ai/resources/research-papers>

IPUs in Research



UNIVERSITY OF BRISTOL TACKLES CHALLENGES IN PARTICLE PHYSICS WITH GRAPHCORE'S IPU

arXiv:2008.09210v1 [physics.comp-ph] 20 Aug 2020

Studying the potential of Graphcore® IPU for applications in Particle Physics

Lakshan Ram Madhan Mohan,^a Alexander Marshall,^a Samuel Maddrell-Mander,^{a,b} Daniel O'Hanlon,^a Konstantinos Petridis,^a Jonas Rademacker,^a Victoria Rege,^b and Alexander Titterton^b

^aH H Wills Physics Laboratory, University of Bristol, UK

^bGraphcore, Bristol, UK

E-mail: lakshan.madhan@bristol.ac.uk, alex.marshall@bristol.ac.uk, sam.maddrell-mander@bristol.ac.uk, daniel.ohanlon@bristol.ac.uk, konstantinos.petridis@bristol.ac.uk, jonas.rademacker@bristol.ac.uk, alexandert@graphcore.ai, victoriar@graphcore.ai

ABSTRACT: This paper presents the first study of Graphcore's Intelligence Processing Unit (IPU) in the context of particle physics applications. The IPU is a new type of processor optimised for machine learning. Comparisons are made for neural-network-based event simulation, multiple-scattering correction, and flavour tagging, implemented on IPUs, GPUs and CPUs, using a variety of neural network architectures and hyperparameters. Additionally, a Kálmán filter for track reconstruction is implemented on IPUs and GPUs. The results indicate that IPUs hold considerable promise in addressing the rapidly increasing compute needs in particle physics.



<https://www.graphcore.ai/resources/research-papers>

IPIUs in Research



IMPERIAL COLLEGE LONDON
ACCELERATE CLASSICAL COMPUTER
VISION PROBLEM ON IPU

Play Video

Bundle Adjustment on a Graph Processor
Joseph O'Neil*, Mark Popel†, Stefan Lorenzberg*, Andrew A. Chien*
*Imperial College London, Department of Computing, U.K. †Graphcore
j.o'neil@imperial.ac.uk

Abstract
Graph processors such as Graphcore's Intelligence Processing Unit (IPU) are part of the major new wave of general computer architectures for AI, and have a general design with unusual, parallel, computation, distributed, on-chip memory and full state communication bandwidth, which allow breakthrough performance for many AI workloads on arbitrary graphs.

We show for the first time that the classical computer vision problem of bundle adjustment (BA) can be solved on a graph processor using Graphcore's Intelligence Processing Unit (IPU). Our results show that the IPU can solve a real BA problem with 121 features and 1000 points in under 10ms, compared to 100ms for the CPU reference. Further code optimisation will easily reduce this difference to near zero, but we report on the real promise of graph processing for parallel computer vision, general, domain-agnostic, throughput-oriented, general purpose, programmable, and domain-specific workloads. In this paper, we describe the design of our implementation, showing the ability of IPU to efficiently solve a real-world problem, and how it compares to a single arbitrary CPU. Both results are then, hardware metrics are given and performance is compared to a single arbitrary CPU.

1. Introduction
Real-world applications which require a general solution. Spatial 2D capabilities from computer vision to AI workloads, but it is clear that a large pay-off will come from solving more problems in real time on GPUs, CPUs and other hardware accelerators. In this paper, we describe the design of our implementation, showing the ability of IPU to efficiently solve a real-world problem, and how it compares to a single arbitrary CPU. Both results are then, hardware metrics are given and performance is compared to a single arbitrary CPU.

REVISING SMALL BATCH TRAINING FOR DEEP NEURAL NETWORKS
Imperial Machine Learning Group
Imperial College London
London, U.K.
imperial.machinelearning@imperial.ac.uk

ABSTRACT
Modern deep neural network training is typically based on mini-batch stochastic gradient descent. While this use of high data batches increases the model generalization performance and allows a significantly smaller memory footprint, there are still significant challenges to overcome. In this paper, we revisit the problem of revisiting small batch training for deep neural networks, and show that it is possible to achieve a similar performance to large batch training, but with a significantly smaller memory footprint. We show that this is possible by revisiting the problem of revisiting small batch training for deep neural networks, and show that it is possible to achieve a similar performance to large batch training, but with a significantly smaller memory footprint.

1. INTRODUCTION
The use of deep neural networks has recently enabled significant advances in a number of applications, including computer vision, speech recognition and natural language processing, and continues to expand rapidly. This has led to a renewed interest in deep learning research, and in particular, the development of deep learning architectures that are more efficient and scalable. In this paper, we revisit the problem of revisiting small batch training for deep neural networks, and show that it is possible to achieve a similar performance to large batch training, but with a significantly smaller memory footprint.



REVISING SMALL BATCH TRAINING
FOR DEEP NEURAL NETWORKS

GRAPHCORE

TRAINING NEURAL NETWORKS IN LOW-DIMENSIONAL RANDOM BASES

Improving Neural Network Training in Low Dimensional Random Bases
Felix Engelmann
Graphcore Research
London, U.K.
felix@graphcore.ai

Paul Eastman
Graphcore Research
London, U.K.
paul@graphcore.ai

Chris Lamb
Graphcore Research
London, U.K.
chris@graphcore.ai

arXiv:2011.10472v1 [cs.LG] 9 Nov 2020

Abstract
Recent Gradient Descent (GD) has proven to be remarkably effective at optimizing deep neural networks, but requires a large number of parameters. This paper introduces a novel method for training neural networks in low-dimensional random bases. This method is able to achieve similar performance to GD, but with a significantly smaller number of parameters. This method is able to achieve similar performance to GD, but with a significantly smaller number of parameters.

1. Introduction
Recent Gradient Descent (GD) has proven to be remarkably effective at optimizing deep neural networks, but requires a large number of parameters. This paper introduces a novel method for training neural networks in low-dimensional random bases. This method is able to achieve similar performance to GD, but with a significantly smaller number of parameters. This method is able to achieve similar performance to GD, but with a significantly smaller number of parameters.



UNIVERSITY OF BRISTOL SOLVES
SCIENTIFIC PROBLEMS WITH NEW
IPU-BASED AI SYSTEMS

Using the Graphcore IPU for traditional HPC applications
Thomas Lane, Steve McInnes-Smith
Bristol, U.K.
thomas.lane@bristol.ac.uk

Abstract
The Graphcore Intelligence Processing Unit (IPU) is a general purpose processor designed for AI workloads. It is a highly parallel, domain-specific architecture that is designed to accelerate AI workloads. This paper describes the use of the IPU for traditional HPC applications, and shows that it is possible to achieve a similar performance to traditional HPC architectures, but with a significantly smaller memory footprint.

1. Introduction
The Graphcore Intelligence Processing Unit (IPU) is a general purpose processor designed for AI workloads. It is a highly parallel, domain-specific architecture that is designed to accelerate AI workloads. This paper describes the use of the IPU for traditional HPC applications, and shows that it is possible to achieve a similar performance to traditional HPC architectures, but with a significantly smaller memory footprint.



GRAPHCORE ACADEMIC PROGRAMME

Apply at:
graphcore.ai/academic

Test IPU hardware in
the cloud at no-cost



Support letters for
grants & funding



Access to Poplar® &
PopART® software



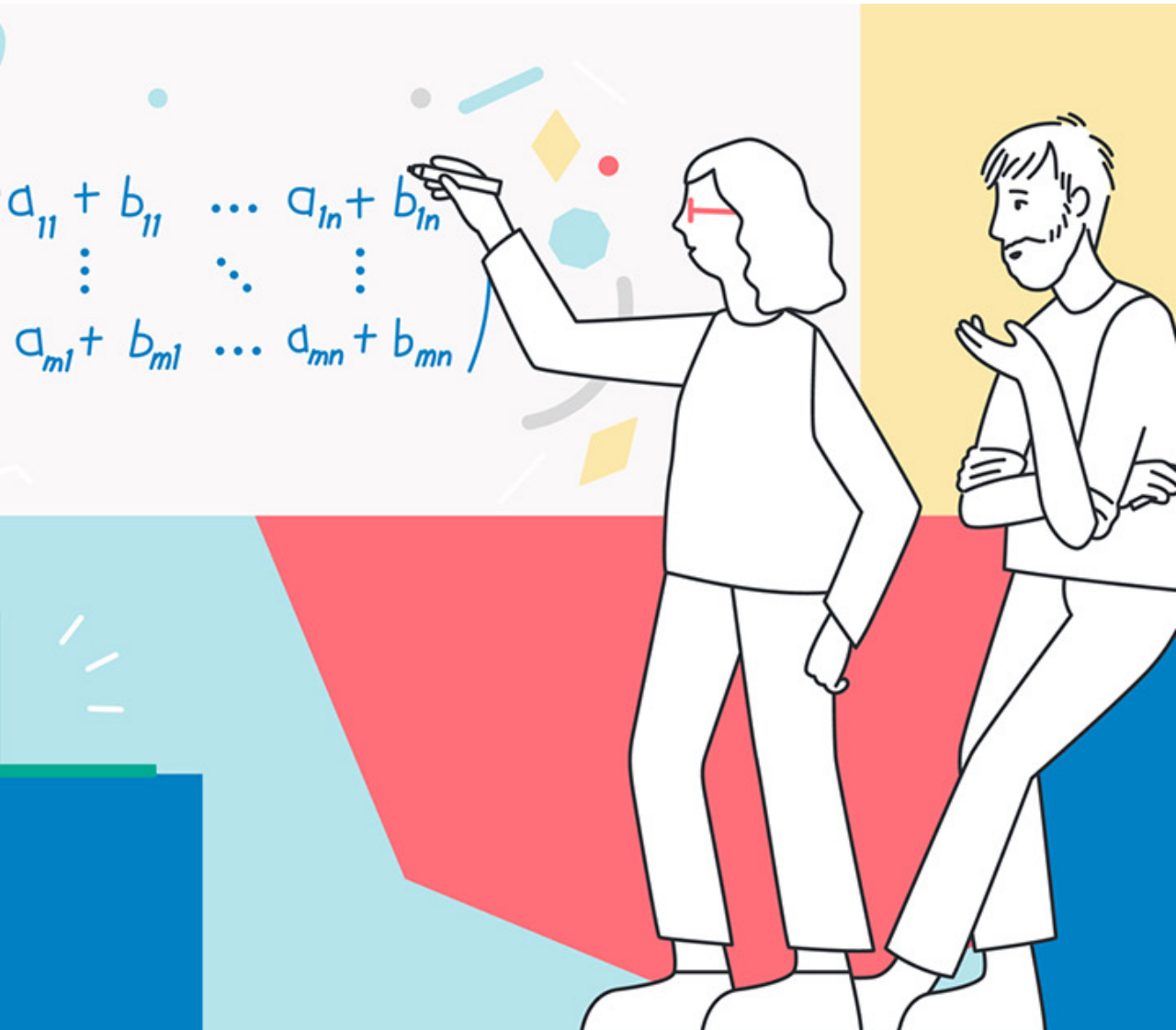
Support from
Graphcore Researchers



RESEARCH PRIORITIES:

- Optimisation of Stochastic Learning
- New Efficient Models for Deep Learning & Graph Networks
- Sparse Training
- New Directions for Parallel Training
- Local Parallelism
- Multi-Model Training
- Conditional Sparse Computation

IPU BENEFITS



Improving ML model performance

Better performance per dollar

Ability to innovate with new models, previously not feasible on legacy architectures

Easy integration with major ML frameworks like TF and Pytorch

Disaggregated, modular architecture that allows for more flexible scale out solutions

**WE HAVE DEVELOPED A NEW KIND OF HARDWARE
THAT WILL LET INNOVATORS CREATE THE
NEXT GENERATION OF MACHINE INTELLIGENCE**



THANK YOU

