

Deep Distributed Intelligence

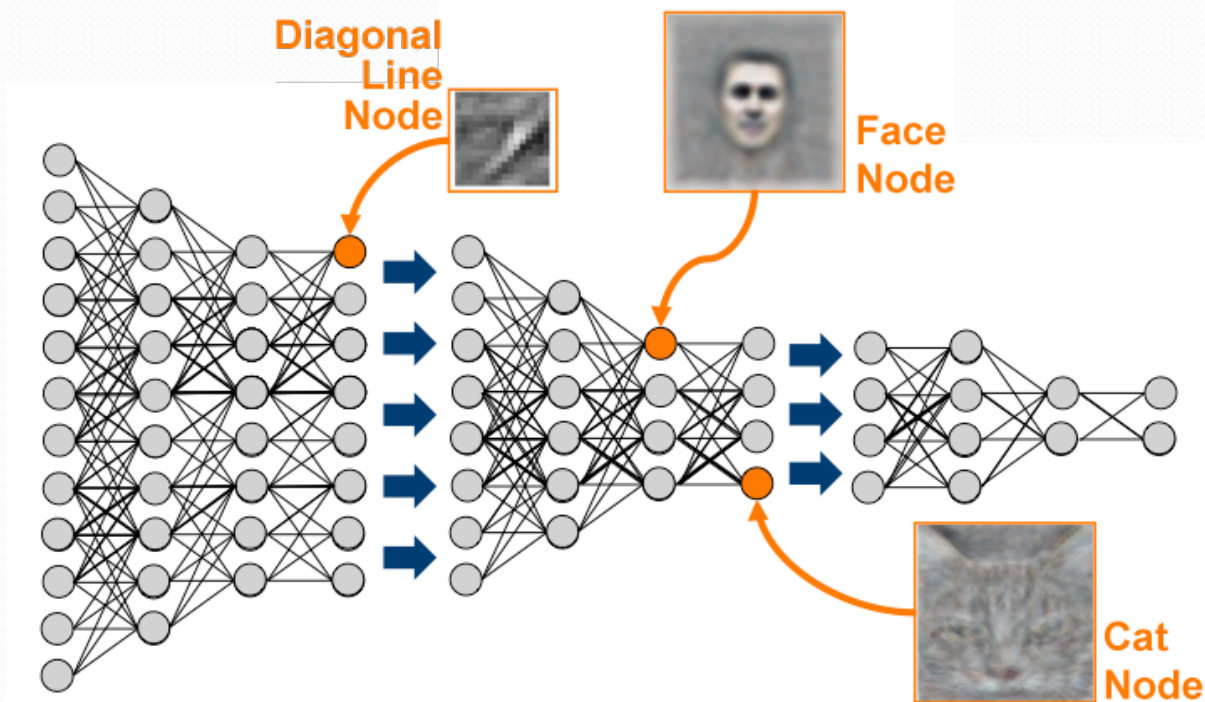
Shaowei Lin (SUTD, ESD)

13 Jul 2016

WNDS Group

Cat Videos

- 2012 Experiment by Google, Stanford (Andrew Ng)
- 3 days, 1000 machines, 16,000 cores, 9-layered neural network, 1 billion connections, 10 million YouTube thumbnails



Le, Quoc V. "Building high-level features using large scale unsupervised learning." IEEE ICASSP, 2013.

MIT Tech Review, "10 Breakthrough Technologies 2013." <http://www.technologyreview.com/featuredstory/513696/deep-learning/>

Speech Translation

- EN Speech => EN Text => CH Text => CH Speech



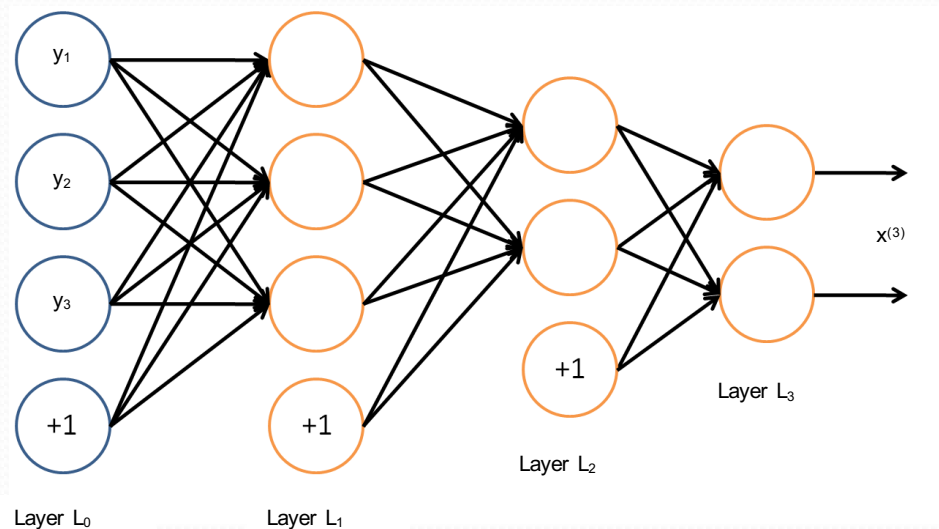
Deep Reinforcement Learning

Space Invaders



What is Deep Learning?

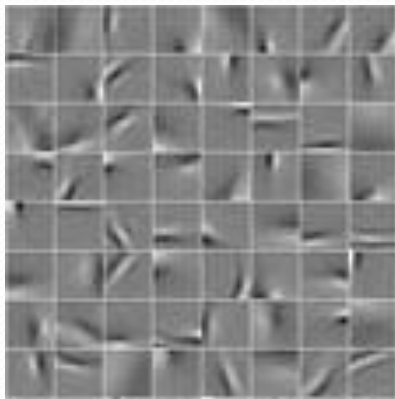
- Biologically-inspired multilayer neural networks



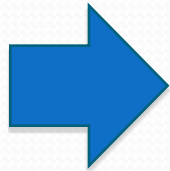
- Unsupervised learning (data without labels)

What is Deep Learning?

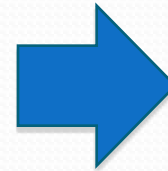
Example. Face recognition (Facebook)



Edges



Eyes, Noses, Mouths

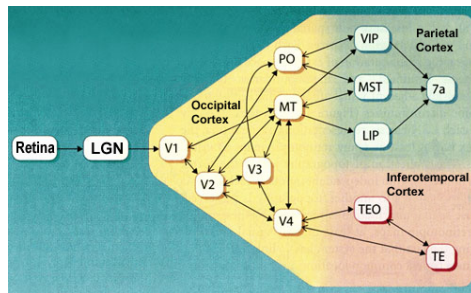


Faces

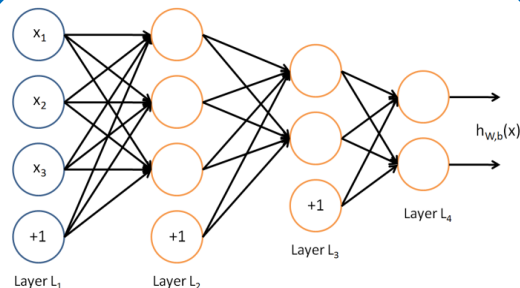
- Deeper layers learn higher-order features

Why Deep Learning?

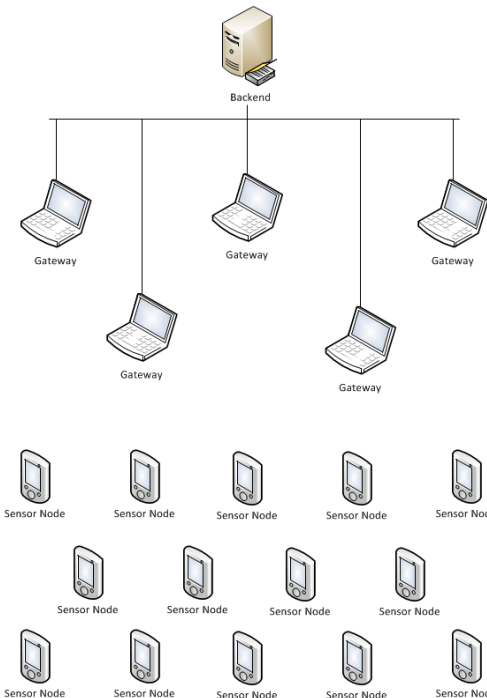
- Sensor networks form the nervous system of smart cities



Deep visual cortex



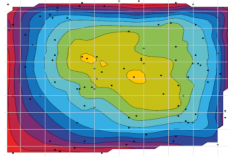
Deep learning



Sensor networks

Deep Learning for Sensor Networks

- Unsupervised learning of features in sensor data

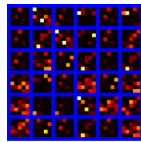


After sparse coding

**Missing Data,
Sparse Sensor
Deployment**



**Smart Sensors
(e.g. Plant Health,
Crowd Sensing)**

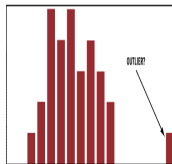


Compressed – 25
bytes

**Data Compression
for Transport
and Storage**



**Fault Detection,
Predictive
Maintenance**



**Outlier Detection,
Multimodal Analysis**



**Intruder Detection,
Data Anonymization**

One learning algorithm. Many inter-connected applications.

Distributed Intelligence

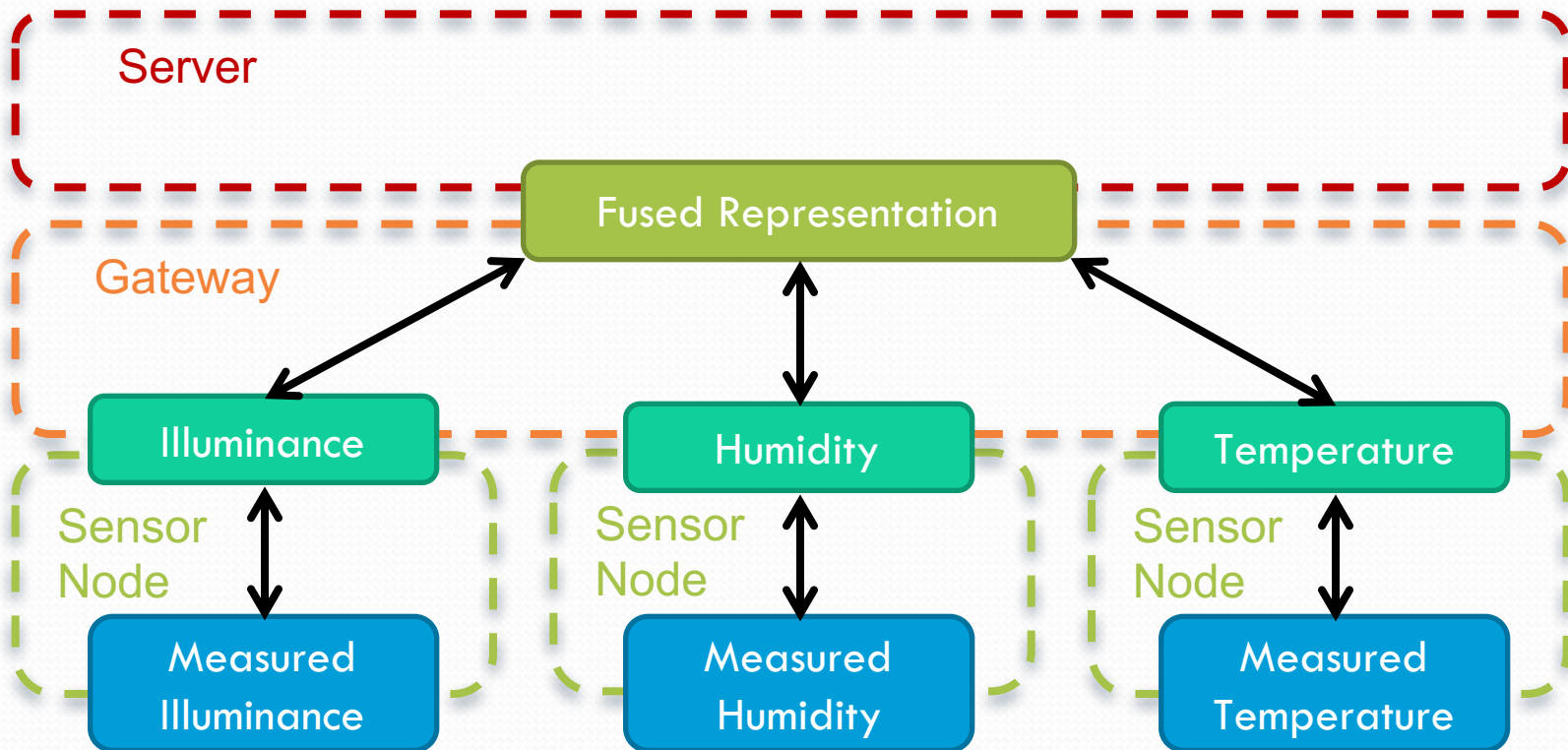
- Fog intelligence, intelligence at the edge
 - Resource-constrained devices
- Information-centric networking
 - Named data networking
 - Moving compute to data
- Decentralization
 - Blockchain
- Real-time streaming
 - Message-passing platforms

Distributed **Intelligence**

- Distributed learning, parallel computing
 - Multi-GPUs, multi-cores, multi-machines
- Unsupervised learning
 - Avoid hand-crafted features
- Online learning
 - Network improves over time
- Data compression. Model compression.
 - Reduce comms, storage, compute needs

Example: Multimodal Deep Learning

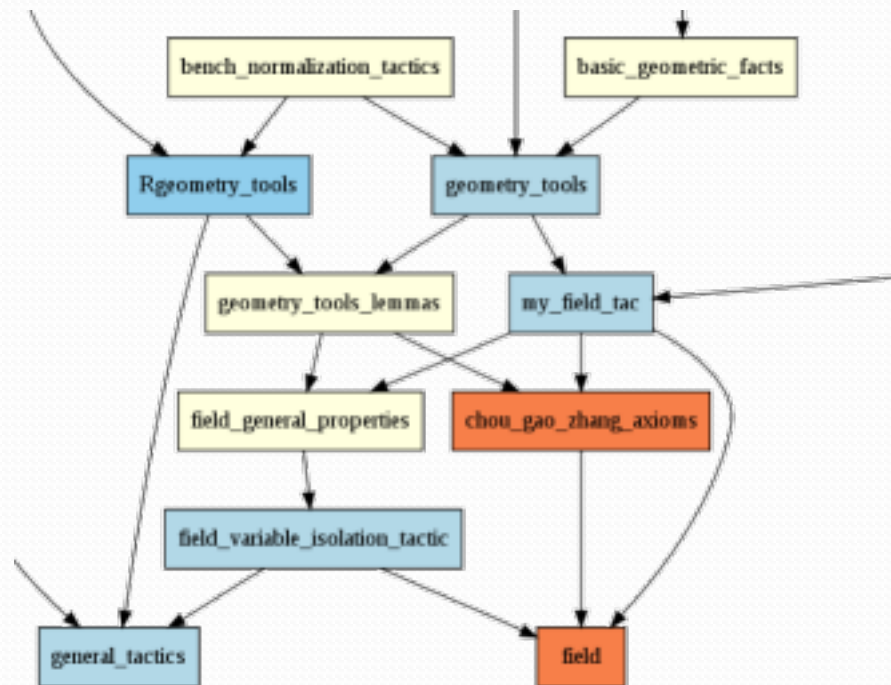
Joint work with Zuozhu Liu and Tony Quek



Step 1: Functional Program

Pipeline of functions

- Computational flow
- Easy to parallelize



Step 1: Functional Program

Type theory

- Logical correctness
- Describe the role, not implementation, of a function

Using Σ -types, a sorting function over lists of natural numbers can be given the type:

```
sort :  $\forall$  (l : list nat), {l' : list nat | sorted l' /\ same_elements l l'}
```

where `sorted` is a predicate that expresses that a list is sorted; and `all_elements` says if two lists contain the same elements. On the contrary, a sorting function in a "poor" type system could only be given the following less informative type:

```
sort : list nat -> list nat
```

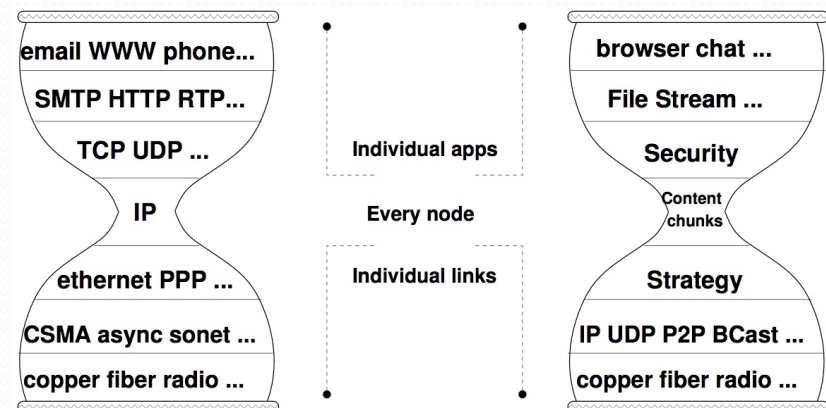
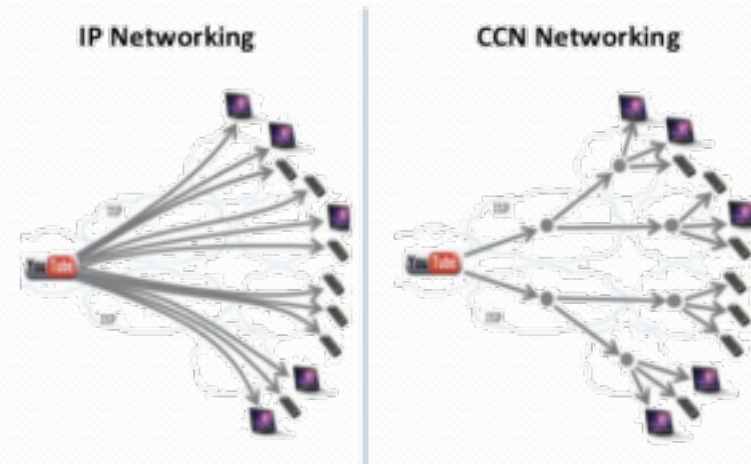
Such a type (specification) enforces the user to write the proofs of predicates `sorted l'` and `same_elements l l'` when writing a implementation for the function `sort`.

Step 2: Data and APIs

Content-centric
networking (CCN)

Named data
networking (NDN)

Dissociation of data
from physical hosts

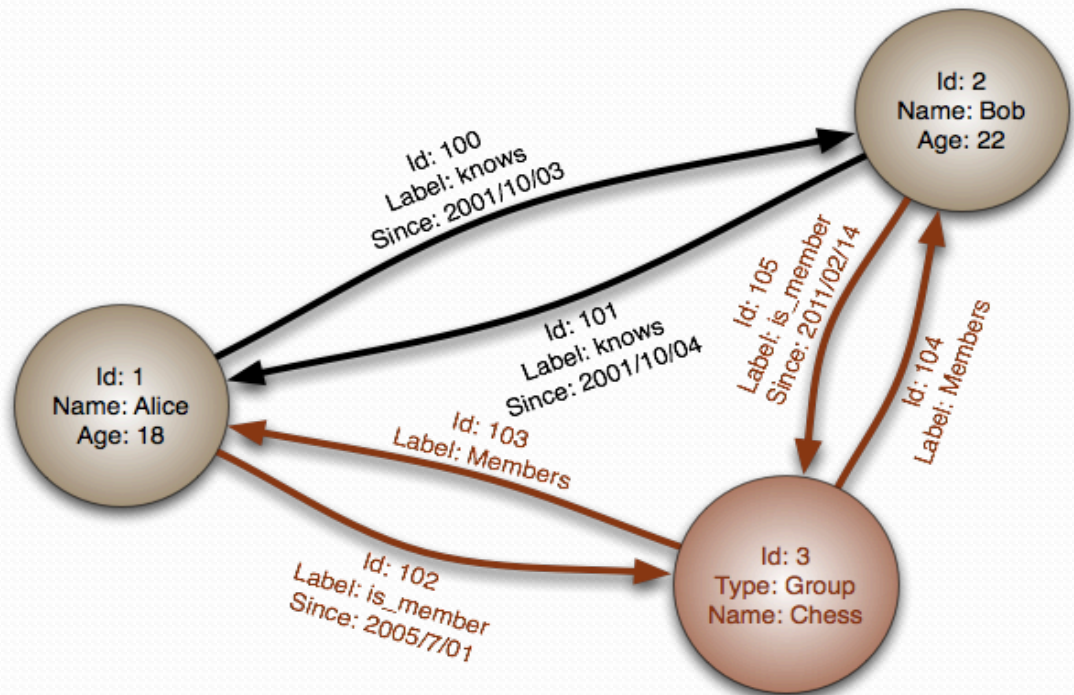


Step 2: Data and APIs

Semantic web

Linked data

Disambiguation

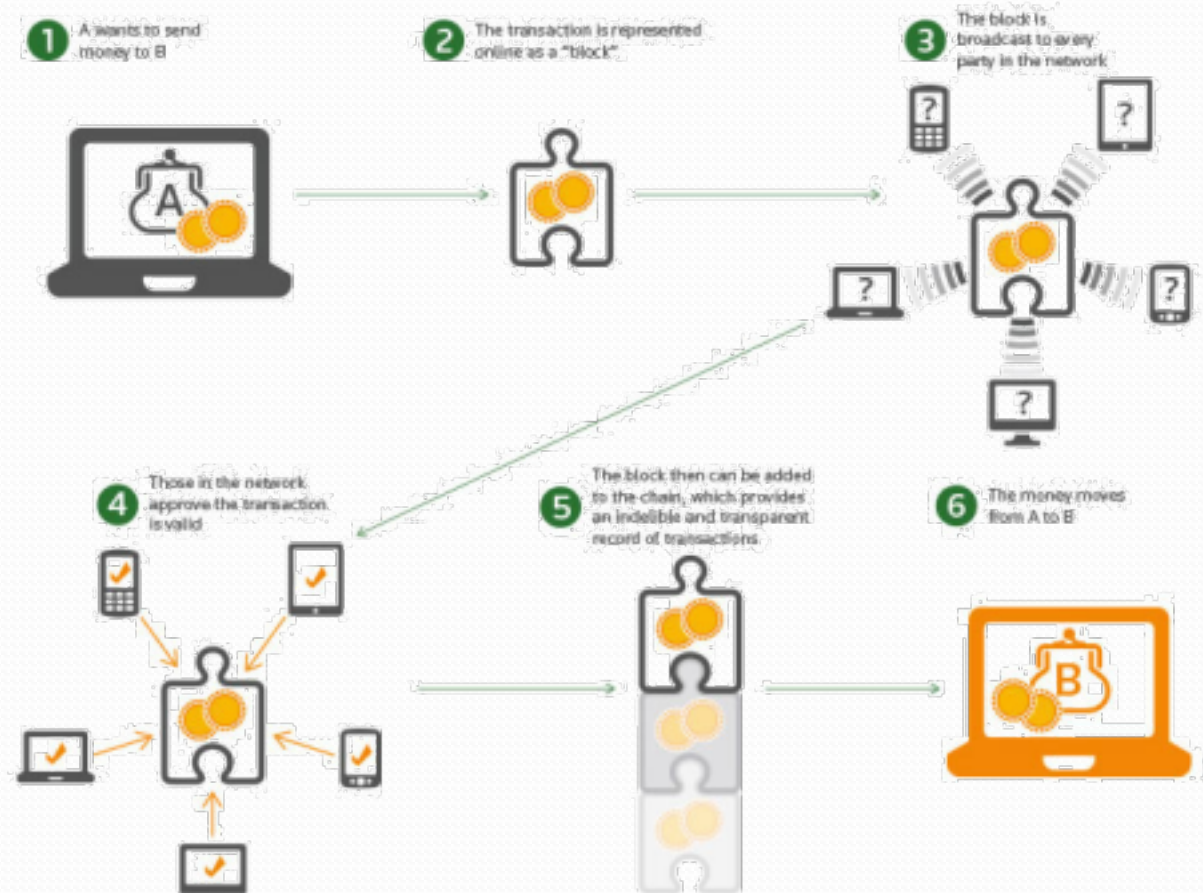


Step 2: Data and APIs

Blockchain vs

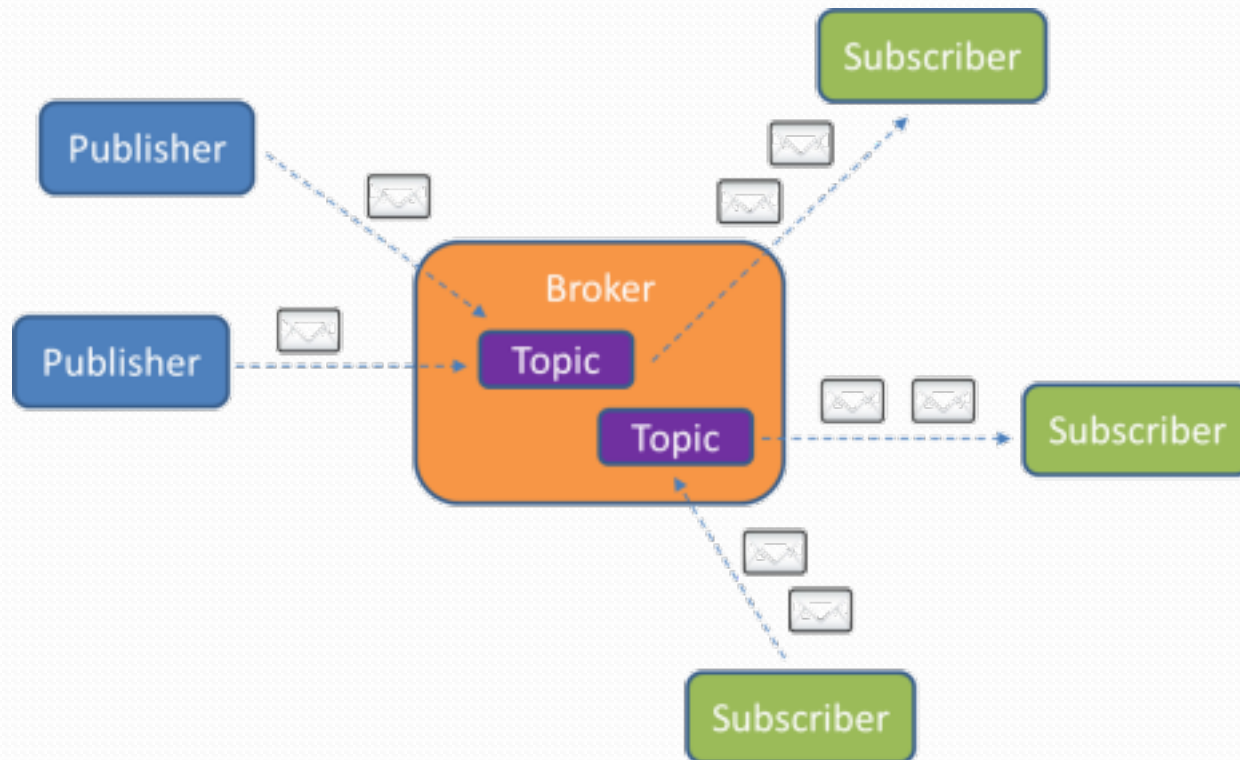
ICANN & DNS

Decentralized
secure database



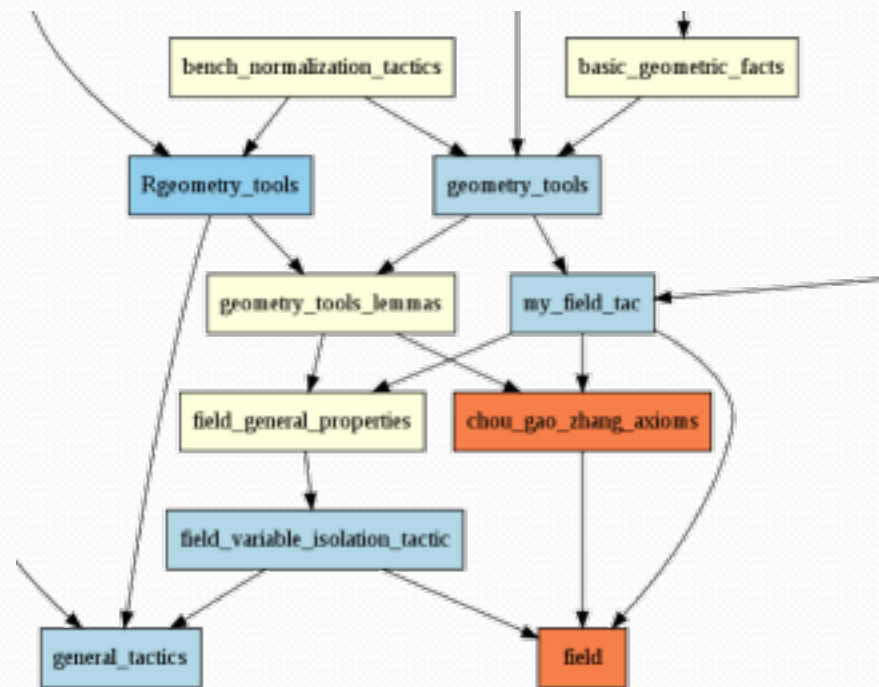
Step 3: Message Passing

Publish-subscribe protocols, e.g. MQTT (IBM).



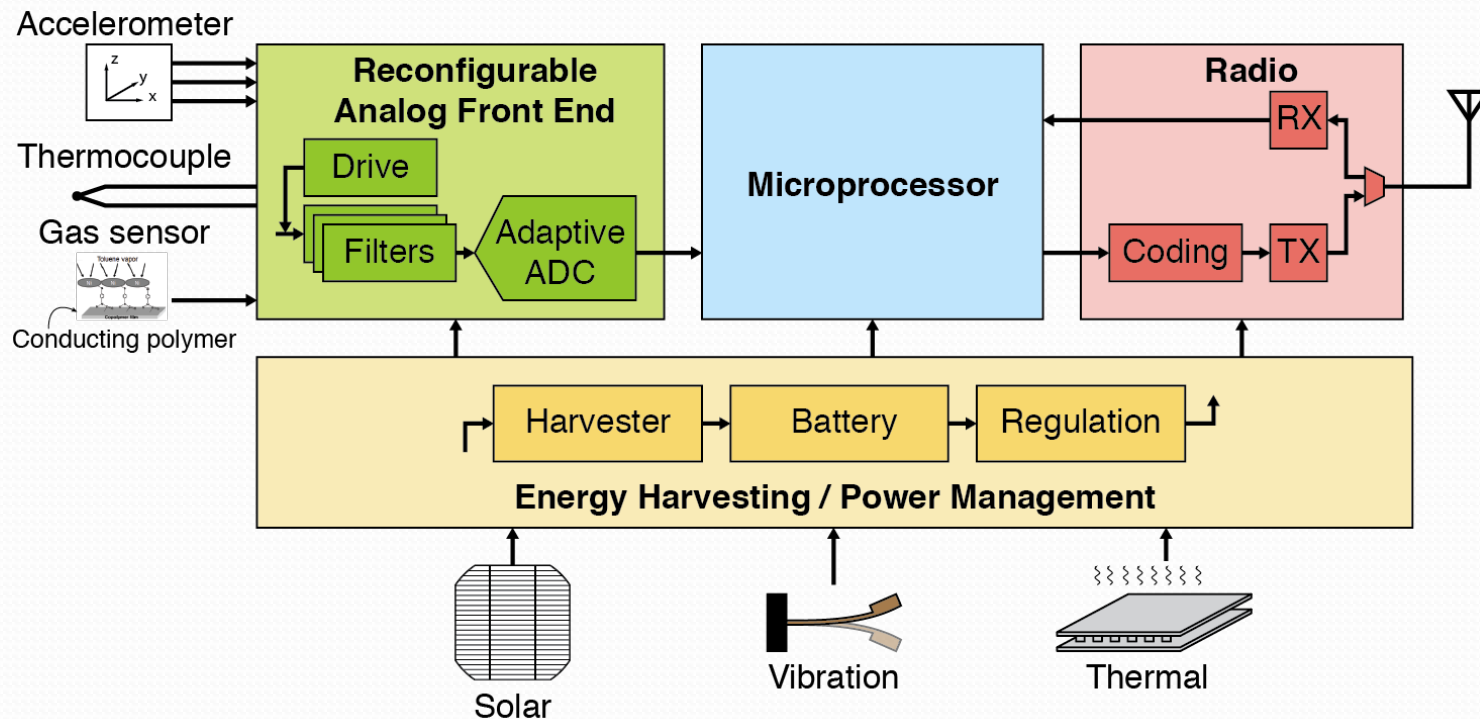
Step 3: Message Passing

Managing real-time information flow through pipelines



Step 3: Message Passing

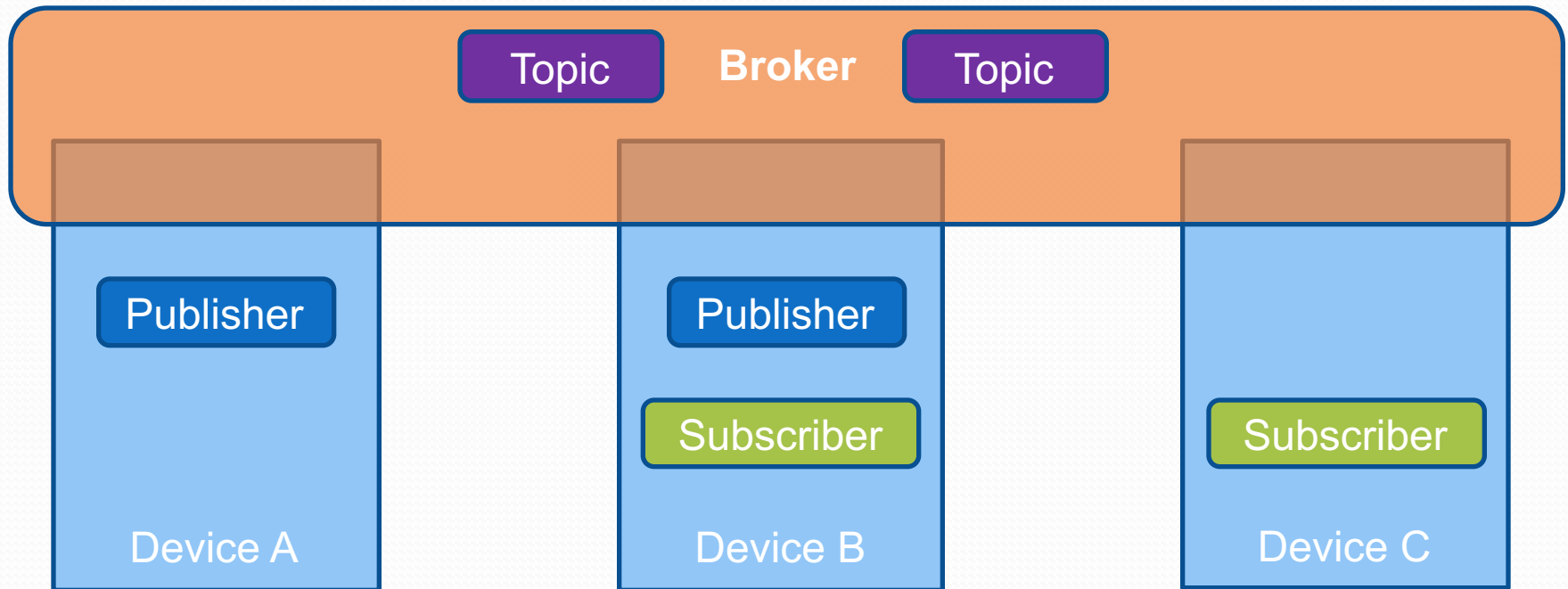
Initiating and terminating processes at run-time.



Joint work with Derek Leong et. al. (I2R)

Step 3: Message Passing

Lightweight distributed broker for seamless cooperation

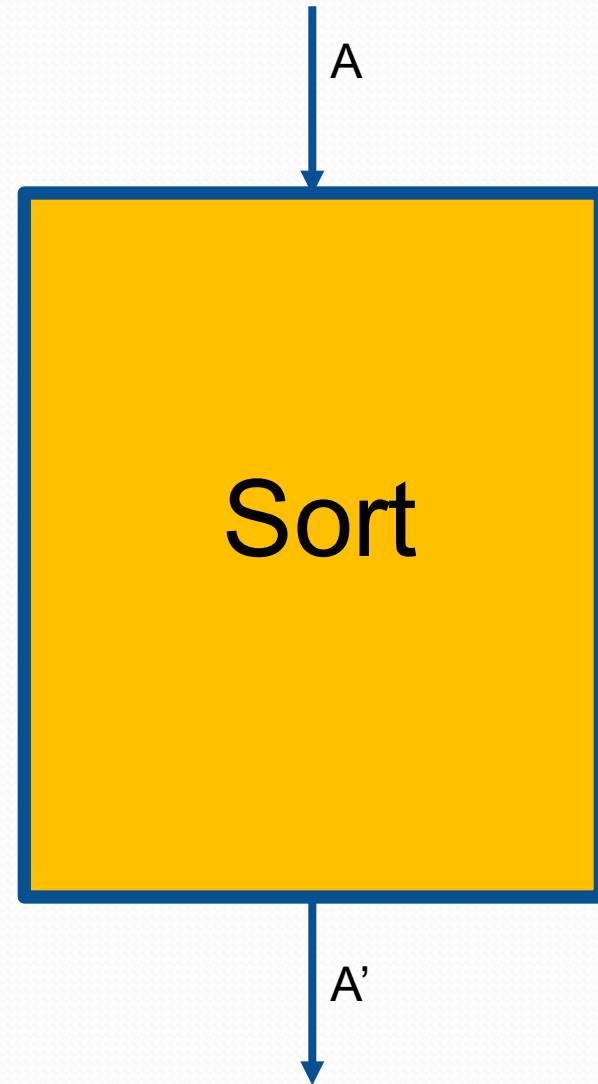


Joint work with Derek Leong et. al. (I2R)

Step 4: Compiling

Converting
functional pipelines
into device primitives
that use physical
resources efficiently.

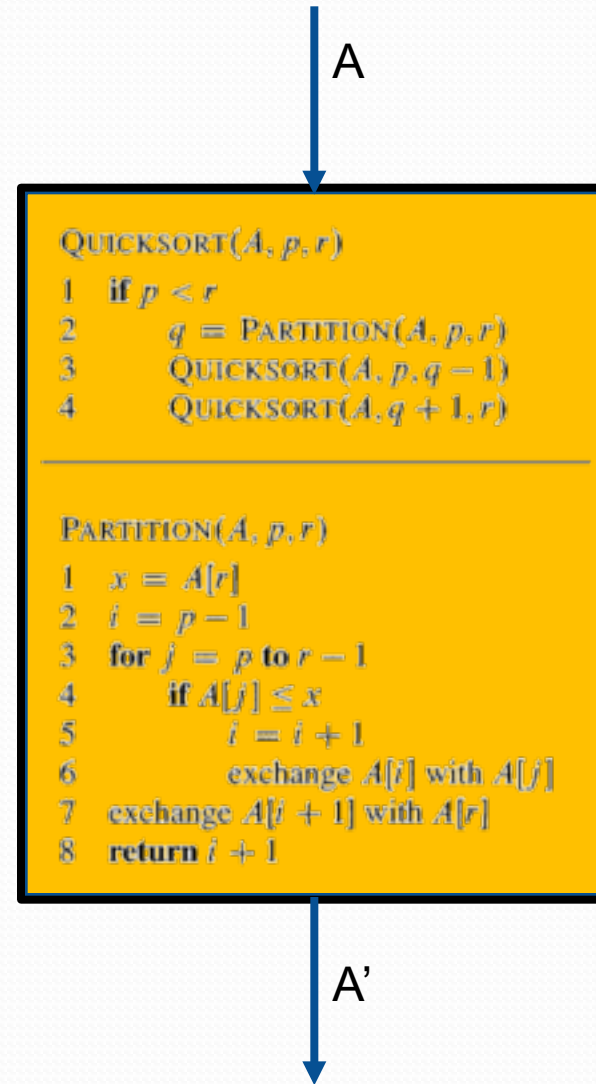
Device dependent
compilation



Step 4: Compiling

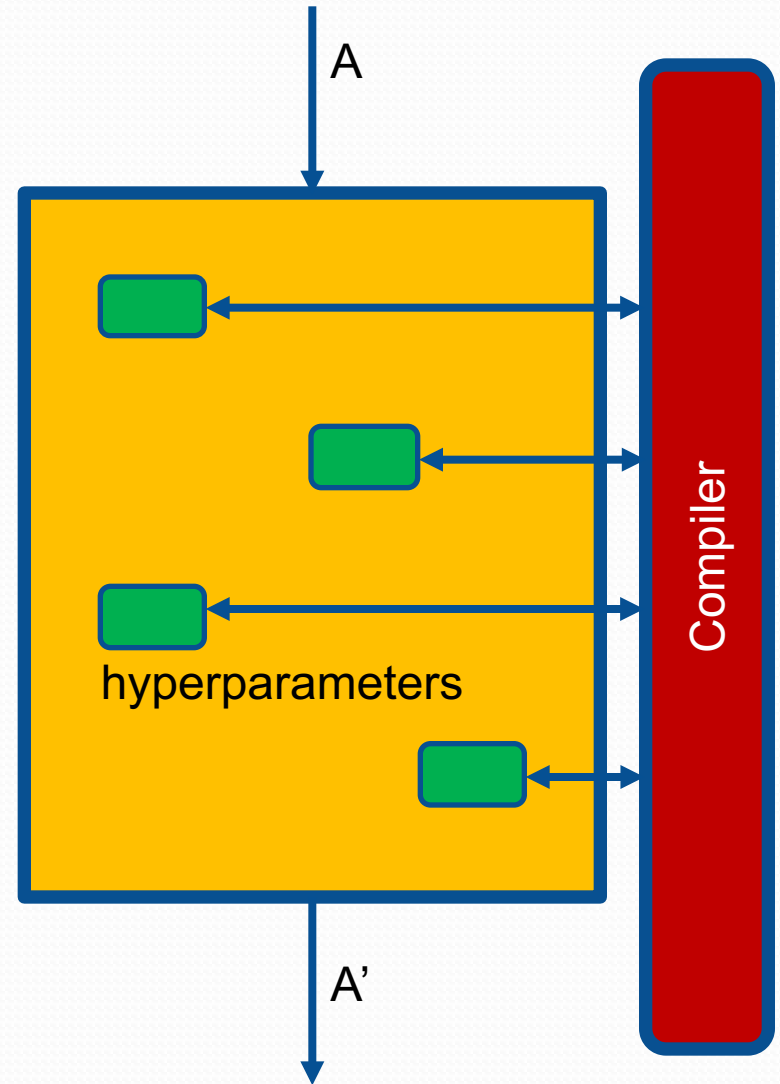
Machines can find functions with similar *type* and suggest how a new function may be implemented.

Tactics in
Proof Assistants



Step 4: Compiling

In self-organizing and self-healing networks, compilers may optimize resources on the fly, with help from **deep reinforcement learning**.



AI = Logic + Learning

Energy efficiency

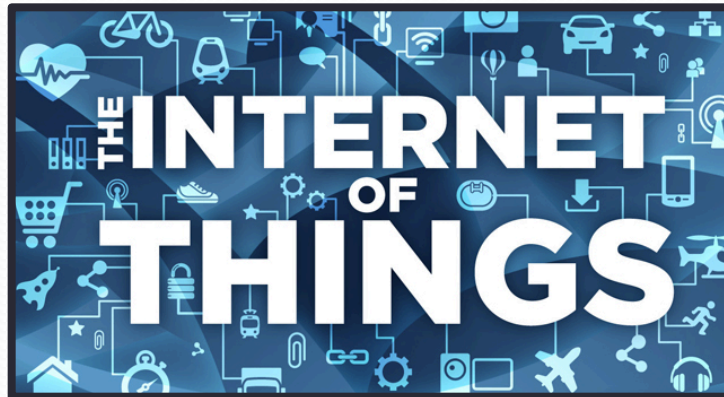
Reliable streaming

Virtual- ization

Inter- operability

Self-organizing

Secure by design



Logic / Language

Learning / Intelligence

Thank you

Lin Shaowei – shaowei_lin@sutd.edu.sg