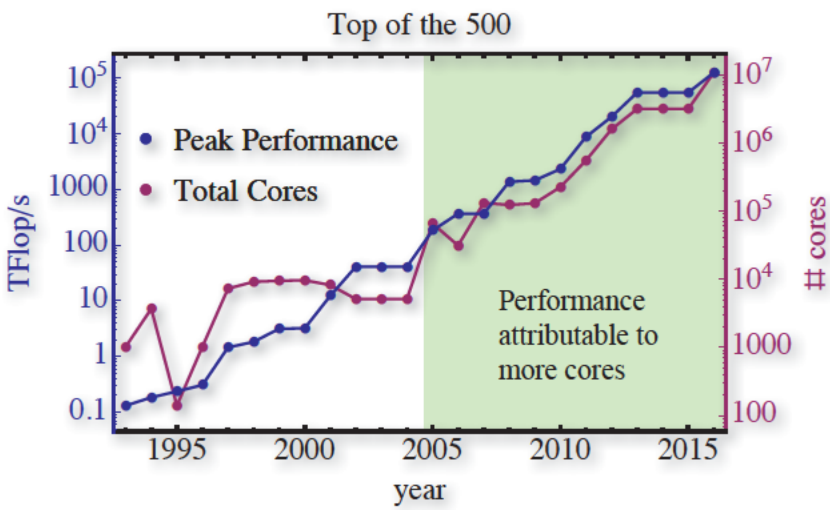
National Aeronautics and Space Administration

Advanced Computing Experiences

From Knights Landing to Quantum Computing

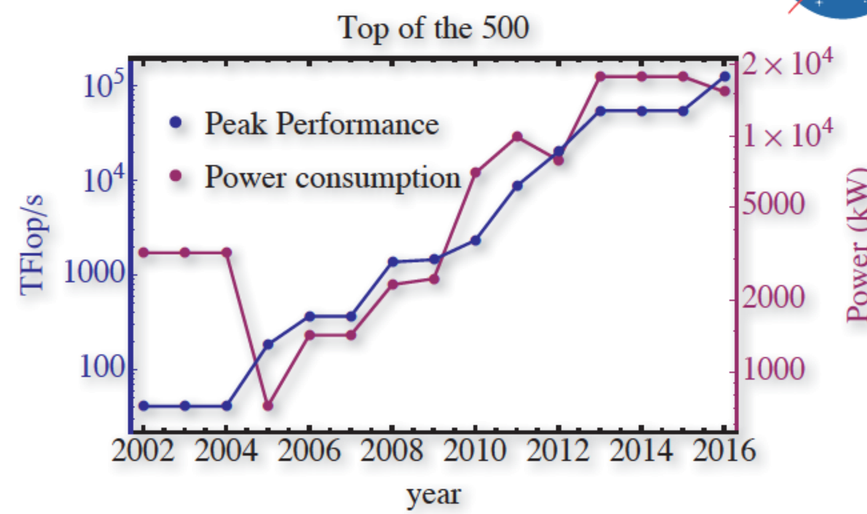
[1,2]Craig Pelissier, [1]Tom Clune, [1]Dan Duffy, [1]Jacqueline Le Moigne
,[1]NASA Goddard Space Flight Center, Maryland, USA
[2]Science Systems and Applications Incorporated, Maryland, USA

The last 20 years …



Getting performance in the last decade:

- Requires fast low-lat¬ncy int¬rconnection fabrics.
- Algorithms with a high degree of parallelism and minimal communication (including low-level vectorization).
- Efficient memory access and caching.
- Efficient communication software: MPI, OpenMP, …
- Limited I/O in and out (parallel file system).



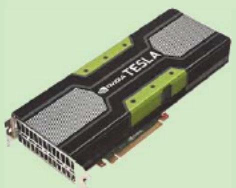
If you ran the current #1 ranked computer (Sunway TaihuLight) **continuously for 1 year …**

= 0.4 % of Utah's energy consumption in 2014.

= Drive a 2015 Tesla S 85D 15.9 x around the Earth's circumference.

Emerging Architectures

Nvidia GPU



- **#3 ranked** Titan Nvidia GPU accelerated computer.
- 52 GPU based computers in the top 500.
- Massively parallel, high latency, large bandwidth.
- Non-caching high-throughput.

KnuPath

- ~biologically inspired microchip architecture.

IBM TrueNorth

- ~ 1 million neurons per chip.
- ~288 million "synaptic" connections.
- Low energy ~70 mW.
- Static chip "no real-time reconfigurability"

Hardware Accelerators, quantum computing, and "brain" inspired microchips.

Intel Xeon Phi

- **#2 ranked** Tiianhe-2 Intel Phi accelerated computer.
- 64 compute cores; L1/L2 cache.
- 16GBs of 3D stacked Multi-channel DRAM 400+ GB/s bandwidth.

D-Wave 2X™

- Quantum Annealing
- Solves QUBOs using quantum effects.
- Highly specialized form of computing.
- First commercially available quantum computer of its size.

Intel Xeon Phi

- Larabee (GPGPU) — 2010
- Knight Ferry (prototype) Co-processor/accelerator — 2010
- Knight Corner (KNC) Co-processor/accelerator — 2012
- Knight Landing (KNL) Co-processor/accelerator Bootable node — 2016
- Knights Mill ? — 2017

2010 | 2011 | 2012 | 2013 | 2014 | 2015 | 2016 | 2017

- Beckton
- Westmere
- Sandy Bridge (SND)
- Ivy Bridge
- Haswell (HSW)
- Broadwell
- Skylake

Lots of resources and large investments …

MilkyWay-2 --- NSCCG

- **#2 54.9 Pflop/s**
- 16,000 Ivy Bridge + 48,000 KNL **co-processors**.

(Gerti) Cori --- NERSC

- **27.9 Pflop/s (NSCCG)**
- 1,630 Ivy Bridge + 9,304 KNLs **(bootable) nodes**.

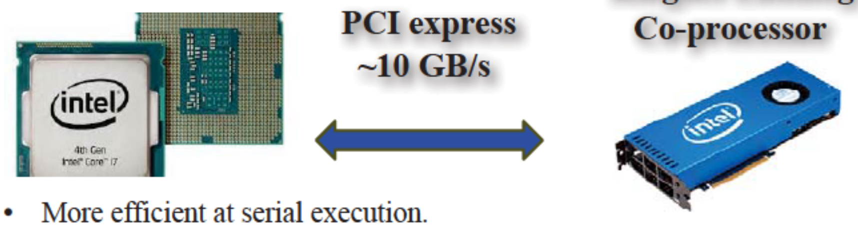
Aurora --- ANL (2018)

- DOE ~$200 million,
- ~100 Pflop/s
- Knights Mill? Deep learning?

?

Knight's Landing Architecture

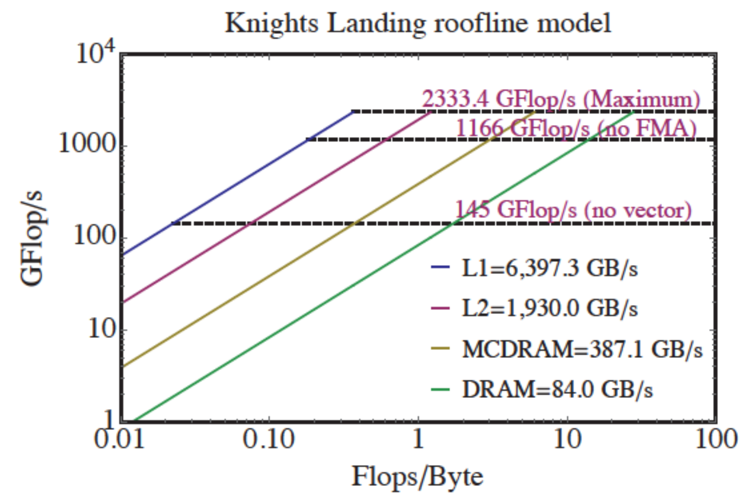
Haswell/Broadwell | **PCI express ~10 GB/s** | **Knights Landing Co-processor**



- More efficient at serial execution.
- I/O (GPFS, Lustre, …)
- Connect to interconnection fabric

- Acceleration of parallel workload

Attribute	Specification
Silvermont 1.4 GHz core	64-72 cores
L1 cache	6,397.3 GB/s
L2 cache	19,300 GB/s
MCDRAM	16 GB > 387.1 GB/s
Hardware threads	4 x 64-72 = 256-288 threads
DRAM	128 GB
Vector processing unit	2 x 512b per core



- **Powerful peak performance ~2.3 TFlop/s --- can you get it?**
  - L1 ~ 0.35 Flops/Byte
  - L2 ~ 2.0 Flops/Byte
  - MCDRAM ~ 6.0 Flops/Byte
  - DRAM ~ 27.7 Flops/Bytes
- **Arithmetic intensity of common routines (double precision)**
  - Vector addition ~ 1/24 Flops/Byte
  - 3D central difference ~ 7/64 Flops/Byte
  - Complex FFT ~ 0.1 Log (N) Flops/Byte
- **Do bootable nodes make sense? What about the serial portion? Is it fundamentally different than the Xeon HPC line?**

Programming on the KNL

Expressing Parallelism

- MPI/OpenMP
- Best performance with MPI+OpenMP --- MPI alone gives less performance.
- OpenMP requires inserting LOTS of directives around parallel regions of the code.

```
!$omp parallel do default(shared)
      do k=1,npz
         call c_sw(delpc(isd,jsd,k), delp(isd,jsd,k),  ptc(isd,jsd,k),    &
                   pt(isd,jsd,k),    u(isd,jsd,k),     v(isd,jsd,k),      &
                    w(isd,jsd,k),   uc(isd,jsd,k),    vc(isd,jsd,k),      &
                   ua(isd,jsd,k),   va(isd,jsd,k), omga(isd,jsd,k),       &
                   ut(isd,jsd,k),   vt(isd,jsd,k), divgd(isd,jsd,k),      &
                   flagstruct%nord,   dt2,  hydrostatic,  .true., bd,     &
                   gridstruct, flagstruct)
      enddo
```

Barrier to entry --- State-of-the-art OpenMP implementation

Optimization --- the usual suspects (same as Xeon architecture)

- Expose as much parallelism as possible
  1. High level (MPI across nodes).
  2. Intermediate level --- thread parallelism.
  3. Low level --- vectorization
- Memory layout, efficient use of caching, working within the MCDRAM.
- Minimize OpenMP forking-and-joining.

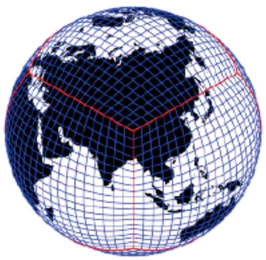
```
!$OMP parallel do default(shared) &
!$OMP& private(k,n)
do nk=1,num_blocks*npz

  k = MOD((nk-1),npz) + 1
  n = MOD(INT((nk-1)/npz),num_blocks) + 1

    call c_sw(Coreb(n)%delpc(:,:,k), Atmb(n)%delp(:,:,k),  Coreb(n)%ptc(:,:,k),    &
    Atmb(n)%pt(:,:,k),    Atmb(n)%u(:,:,k), Atmb(n)%v(:,:,k),                      &
    Atmb(n)%w(:,:,k),   Atmb(n)%uc(:,:,k),   Atmb(n)%vc(:,:,k),                    &
    Atmb(n)%ua(:,:,k),   Atmb(n)%va(:,:,k), Atmb(n)%omga(:,:,k),                   &
    Coreb(n)%ut(:,:,k),   Coreb(n)%vt(:,:,k), Coreb(n)%divgd(:,:,k),               &
    flagstruct%nord,   dt2,  hydrostatic,  .true., bdb(n),                         &
    Atmb(n)%gridstruct, flagstruct)

enddo
```

Results FV3

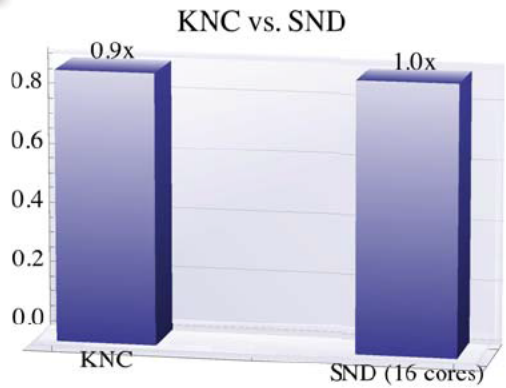


Cubed-Sphere
Finite-volume dynamical-core (3)

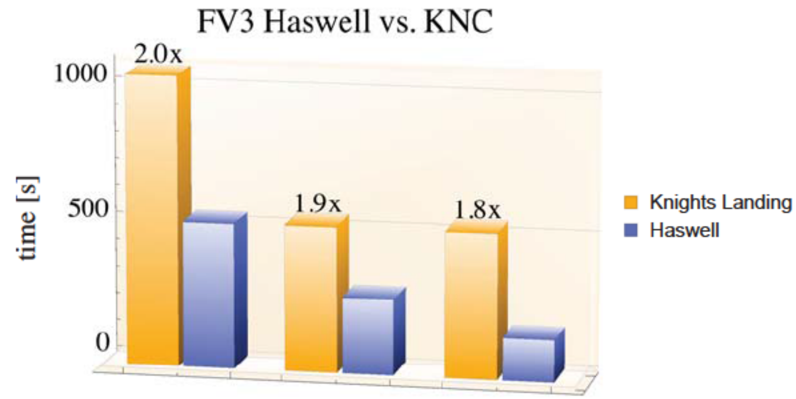
- Key component of GEOS5 flagship software at GSFC.
- Solves fluid dynamics equations (Navier-Stokes).
- One of the most computationally expensive component.
- Shared with other organizations (GFDL, NOAA, NCEP, …).

Knights Corner --- FV2

- Code refactored – blocking
- OpenMP ported including communication routines across nodes with MPI
- 3 people ~9 months.



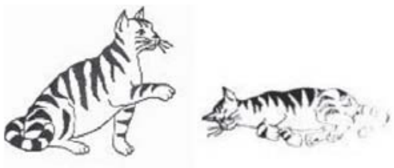
2016 Multi-core Workshop, Boulder, CO



Results taken from Jim Rosinski NOAA/ESRL (2016) presentation.

- **Significant performance boost for Xeon vs. Xeon Phi architectures this time around.**
- **Performance boost "largely" attributable to MCDRAM bandwidth boost ~2x Haswell.**
- **Is Intel putting MCDRAM on Xeon? Intel Xeon Phi fundamentally different than Xeon?**

Quantum Computing



Quantum Annealing

Problems:
Optimization
Boltzmann sampling

"Analog" Quantum Computing

Problems:
Optimization, Material Science, Quantum Chemistry, Sampling, Quantum Dynamics

Universal Quantum Computing

Problems:
All quantum algorithms

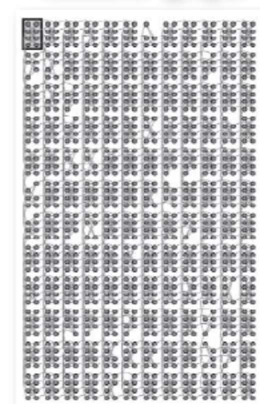
→ Computational power

D-Wave 2X™
NASA Ames Quantum Artificial Intelligence Lab

1152 (8x12x12) qubit "Washington" processor
1097 qubits in "Working Graph"
15 mK Max operating temperature (13 mK nominal) Key feature: A small reduction in temperature provides a significant boost in performance
3.5% and 2% precision level for h and J (couplings
~10xT (~4x improvement of adiabatic process)
Graph connectivity: 6 per qubit (Chimera architecture)

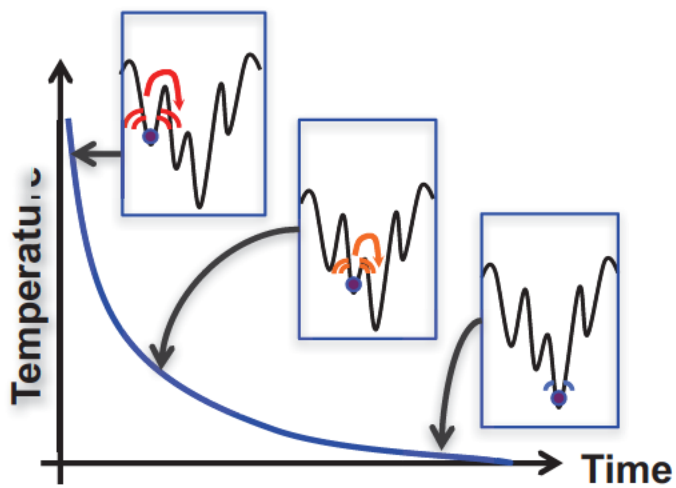
Cooling system
1 mm² chip

"chimera" graph



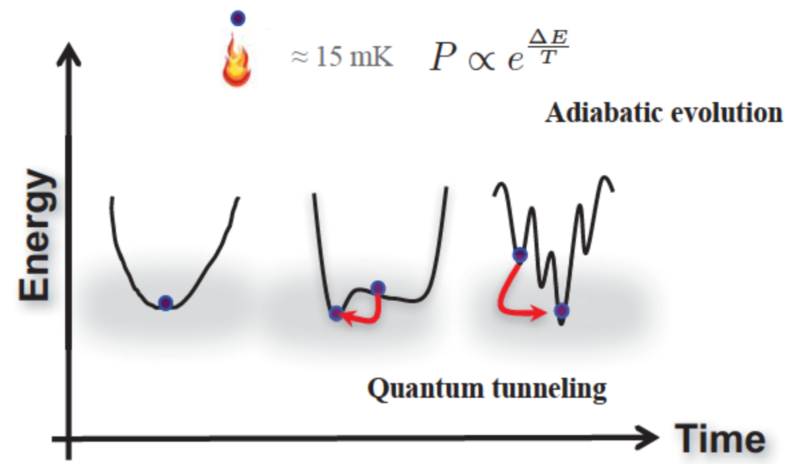
Quantum Annealing

Simulated annealing
(Kirkpatrick et al., 1983)



strategy: Heat up and cool off slowly.
Escape mechanism: Thermal (kinetic) energy
Outcome: Settle into local minima.

Quantum annealing
(Finnila et al., 1994, Kadawaki and Nishimori, 1998, Farhi et.al., 2001)



strategy: Adiabatic evolution.
Escape mechanism: Quantum tunneling
Outcome: Settle into **global** minima.

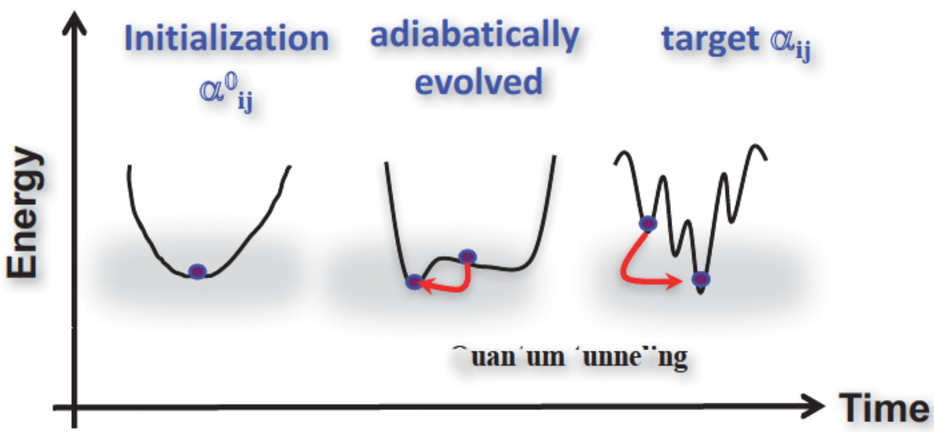
Quantum Annealing on D-Wave

D-Wave can solve QUBOs

$$E = \sum_{i,j} \alpha_{ij}\, q_i q_j$$

Can minimize functions like this!

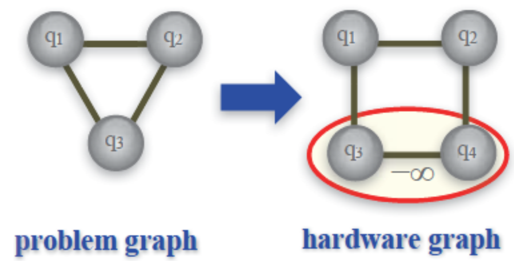
- NP-hard problem --- intractable in classical computers.
- q_i are the "qubits" --- 1097.
- α_{ij} are the qubit connections limited to hardware connectivity.



1. Formulate problem as a QUBO

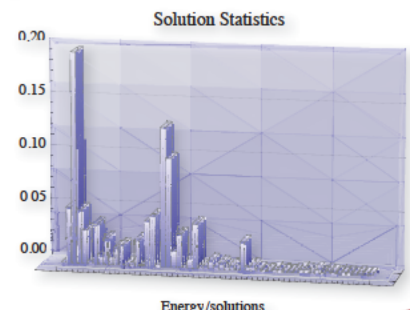
$$E = \sum_{i,j} \alpha_{ij}\, q_i q_j$$

2. Embed the problem in hardware



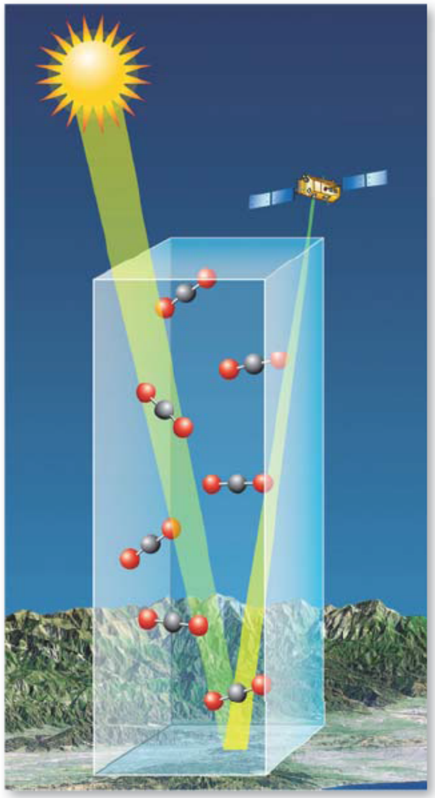
- **Requires more hardware qubits!**
- **Fully connected ~N²**
- **Susceptible to errors!**
- **API solver to do this for you.**

3. Generate statistics and take the best answer



Quantum mechanics is non-deterministic!

Estimating Terrestrial CO_2

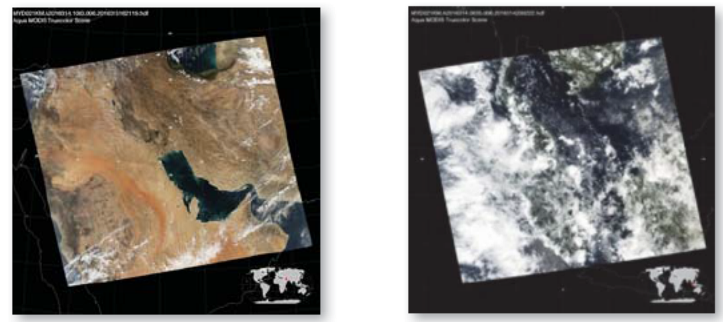


Orbiting Carbon Observatory-2 (2014) measures atmospheric CO_2 around the globe.

Will the impact of "Global Warming" affect the rate of anthropogenic CO_2 uptake by land vegetation?

Moderate Resolution Imaging Spectroradiometer (MODIS)

Collects visible and infrared "pictures" of entire earth surface every 1-2 days.



Estimating vegetation density or "greeness" from MODIS images.

Images need to be **aligned** or "registered".

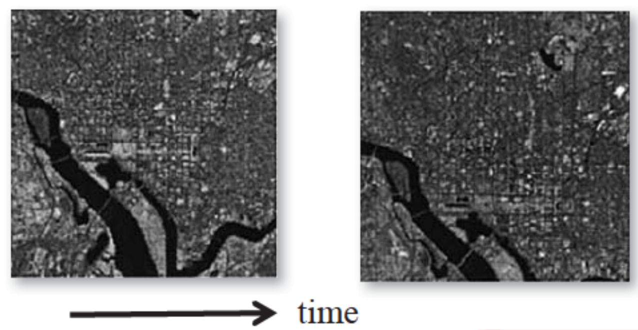


Image Registration

Registration as a QUBO

Use fixed precision for real value transformation parameters (θ, a_x, a_y).

$$x = x_{\min} + \Delta x \sum_{i=1}^{N_b} 2^{i-1} q_i \,, \quad \Delta x = \frac{x_{\max} - x_{\min}}{2^{N_b}}$$

How to do a rotation.

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

$$\rightarrow \begin{pmatrix} c & s \\ -s & c \end{pmatrix}$$

Can't evaluate transcendental functions!

Add constraint $c^2+s^2=1$

$$\lambda(1 - c^2 - s^2)^2$$

How to do translation.

$$\begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} x + a_x \\ y + a_y \end{pmatrix}$$

No constraint needed!

Point-to-point correspondence (All labeling permutations).

$$\begin{pmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{pmatrix}$$

Rows and columns sum to unity.

$$\lambda \sum_i \left(1 - \sum_j q_{ij}\right)^2 + \lambda' \sum_i \left(1 - \sum_j q_{ij}\right)^2$$

Final PUBO (QUBO for small angles)

$$\mathcal{O} = \lambda_4 \sum_i (\mathcal{Q}\mathbf{p}_i - \mathbf{p}'_i)^2$$
→ **Pixel value comparison**

$$+ \lambda_5 \sum_{i=1}^{N} \left[\mathcal{Q}\mathbf{r}_i - \left(\begin{pmatrix} c & s \\ -s & c \end{pmatrix} \mathbf{r'}_i + \mathbf{a}\right)\right]^2$$
→ **Rot. + Trans. violation.**

$$+ \lambda_1 \sum_j \left(\sum_i q_{ij} - 1\right)^2 + \lambda_2 \sum_i \left(\sum_j q_{ij} - 1\right)^2 + \lambda_3 \left(c^2 + s^2 - 1\right)^2 .$$
} **Labeling + geometrical constraints.**

Issues:

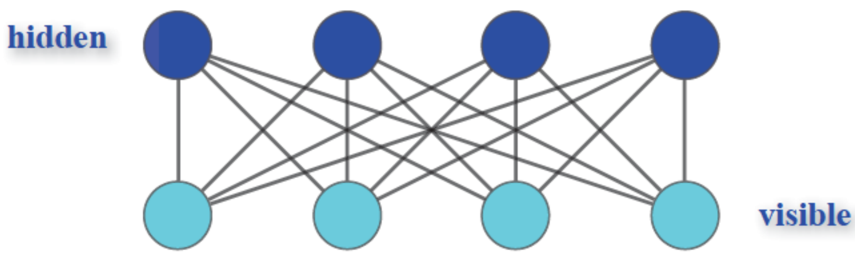
- Requires entire machine to register 6 points.
- Embedding almost always broken down (400/10000).
- Possible to improve accuracy with error correction and careful adjustment of constraint parameters.
- Image registration algorithms perform the same task in polynomial time ~N^2

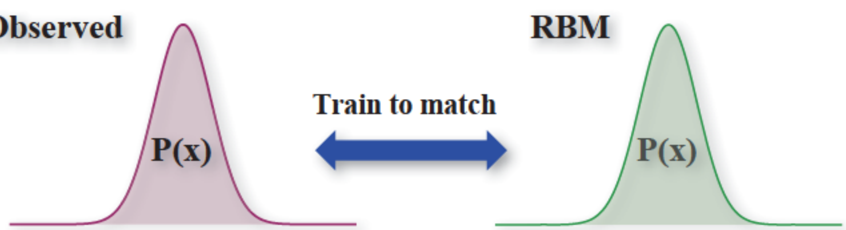
Takeaway:

- **Real valued problems formulated directly as QUBOs are not a good idea.**

Restricted Boltzmann Machines

Bipartite Binary Stochastic Neural Network.

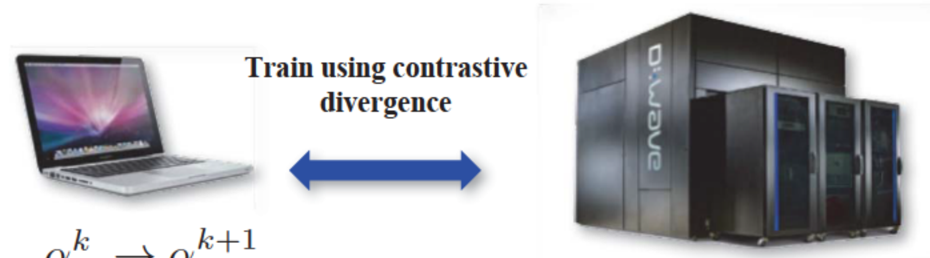




Same distribution as D-Wave!

$$P \propto e^{-E/T} \qquad E = \sum_{i,j} \alpha_{ij}\, q_i q_j$$

D-Wave is a physical RBM!



$$\alpha_{ij}^{k} \rightarrow \alpha_{ij}^{k+1}$$

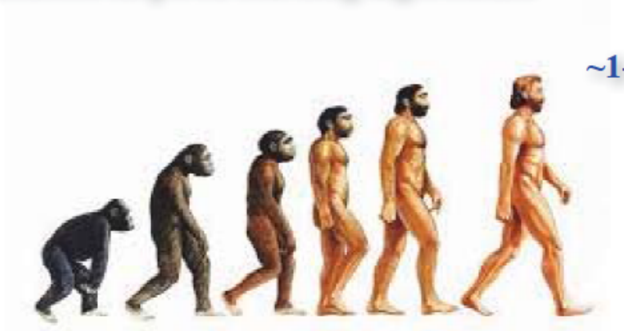
Sample Boltzmann statistics

Can D-Wave produce Boltzmann sampling more efficiently than classical computers?

Feature extraction with RBMs? Alignment?

Artificial Neural Networks

Evolution inspired learning algorithms.



~1-2 million years

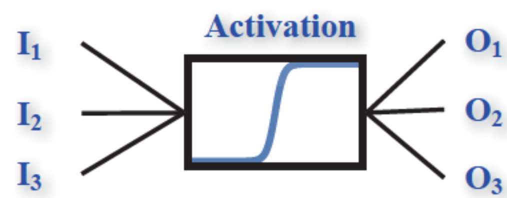
The power of the human brain.

20 – 40 Watts

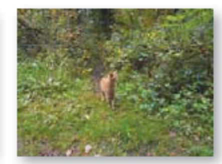
Sunaway TaihuLight

~0.4 – 0.8 million
times more power!
(15,371 kW)

Model as a collection of connected neurons.

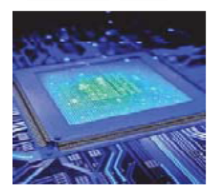


Perform tasks "we" are good at!

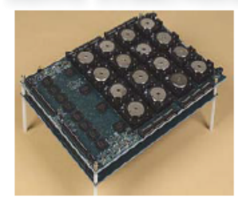


Application specific hardware.

KnuPath



IBM TrueNorth



Thank You!