Scalable and High Performance Computing for Big Data Analytics in Understanding the Human Dynamics in the Mobile Age

Xuan Shi
GRA: Bowei Xue
University of Arkansas

SAN DIEGO STATE UNIVERSITY
KENT STATE UNIVERSITY
UNIVERSITY OF ARKANSAS

Spatiotemporal Modeling of Human Dynamics Across Social Media and Social Networks
Interdisciplinary Behavioral and Social Science Research, National Science Foundation

Overview

- Background and needs of HPC in big data science
- Emerging hybrid computing infrastructure and technologies
- Social media data analytics for understanding the human dynamics in the mobile age
- Current and future initiatives

Background and needs of HPC in big data science

Briefing Room
- Your Weekly Address
- Speeches & Remarks
- Press Briefings
- Statements & Releases
- White House Schedule
- Presidential Actions
  - Executive Orders
  - Presidential Memoranda
  - Proclamations
- Legislation
- Nominations & Appointments
- Disclosures

The White House
Office of the Press Secretary

For Immediate Release July 29, 2015

Executive Order -- Creating a National Strategic Computing Initiative

EXECUTIVE ORDER

\- - - - - - -

CREATING A NATIONAL STRATEGIC COMPUTING INITIATIVE

By the authority vested in me as President by the Constitution and the laws of the United States of America, and to maximize benefits of high-performance computing (HPC) research, development, and deployment, it is hereby ordered as follows:

Background and needs of HPC in big data science

"Theory and experimentation have for centuries been regarded as two fundamental pillars of science. It is now widely recognized that computational and data-enabled science forms a critical third pillar."

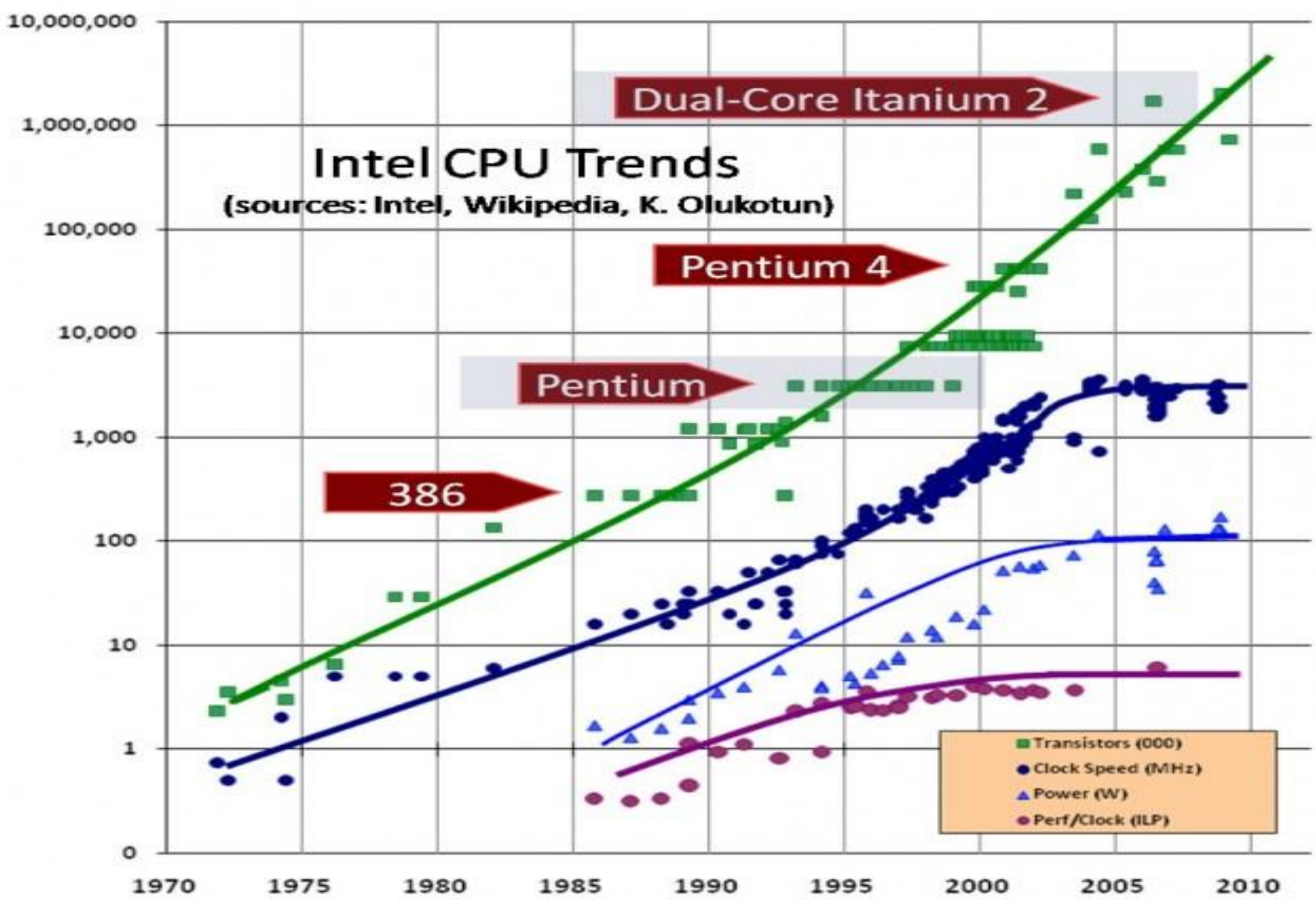
- National Science Foundation

- Spatiotemporal computation is the third pillar of our transdisciplinary research project entitled "*IBSS: Spatiotemporal Modeling of Human Dynamics across Social Media and Social Networks*" sponsored by NSF

Background and needs of HPC in big data science

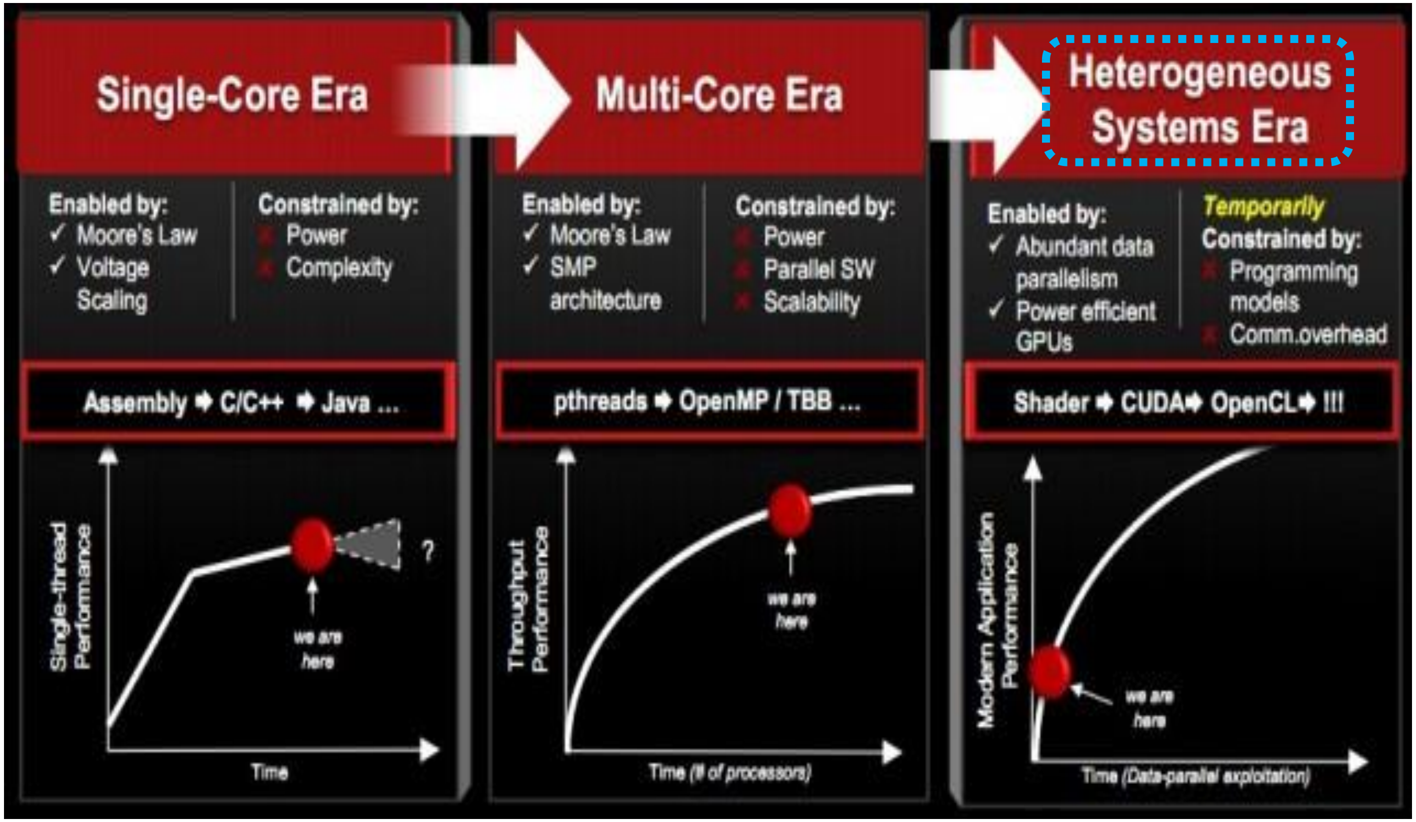
- **Different from the traditional attribute-value data, the vast amount of social media data are largely *unstructured, noisy, distributed, and dynamic*.**
- **This research has to develop effective solutions and tools for data processing, visualization, data mining and analytics to study human dynamics and information diffusion via social media**
- **With the exponential growth of data, the challenge of scalable and high performance computing for big data analytics become *urgent* because many research activities are constrained by the *inability* of software or tool that even could *not* complete the computation process.**
- **Many large-scale geospatial problems may be not processable at all if the computer system does not have sufficient *memory* or *computational power*.**

The trend in hardware advancement



CPU scaling showing transistor density, power consumption, and efficiency. Chart originally from The Free Lunch Is Over: A Fundamental Turn Toward Concurrency in Software

The trend in software development



Multicore + accelerators (e.g. GPUs, MICs)

Graphic processing unit (GPU)

G80 → GT200 → Fermi → Kepler → Maxwell

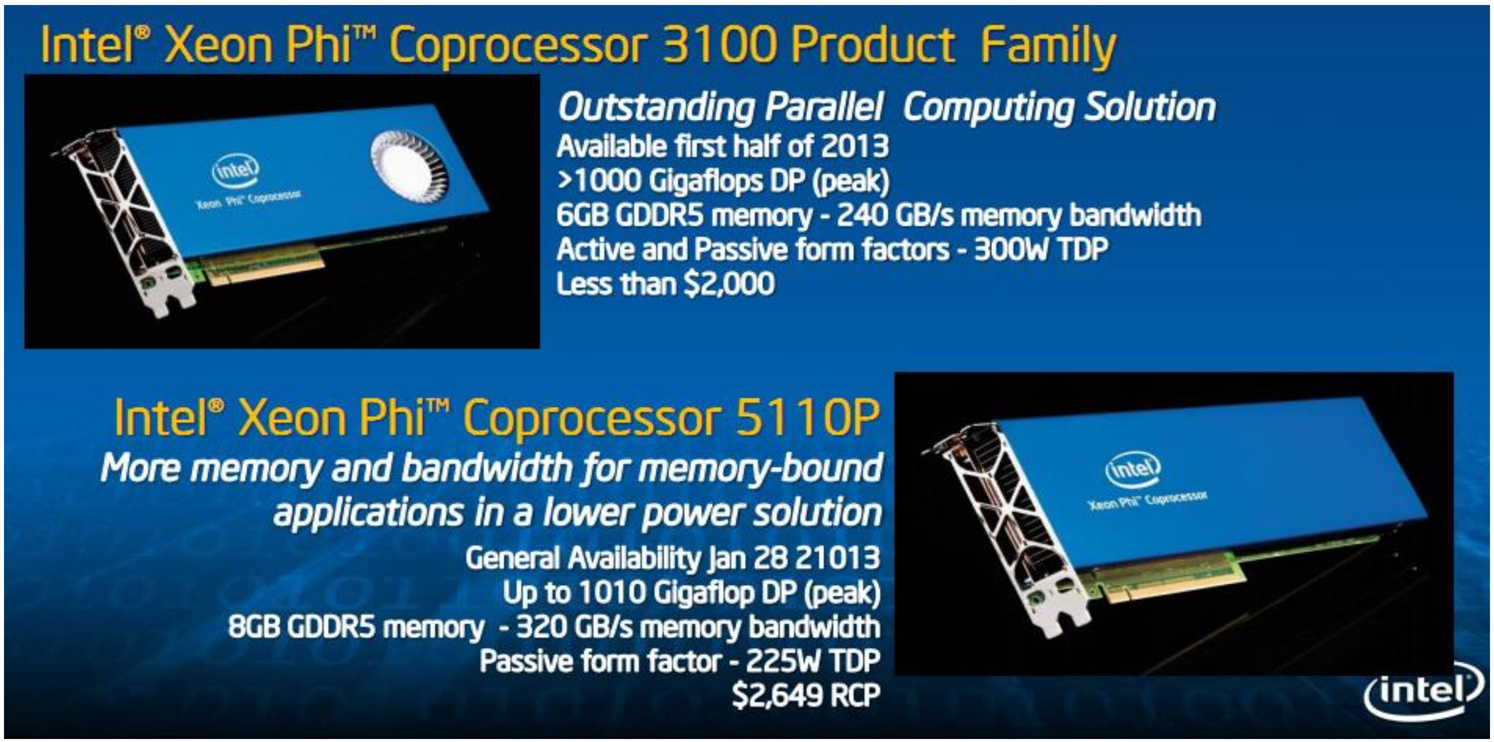
Kepler K20/K20X/K40

13~15 streaming multiprocessors (SMXes)

Each SMX has 192 single-precision cores and 64 double-precision cores

Processing Units	13 Stream Multiprocessors (SMXs), each with 192 cores, 2,496 cores total
Maximum Threads	13 SMXs, each with 192 cores, 32 way SIMD = 79,872 threads

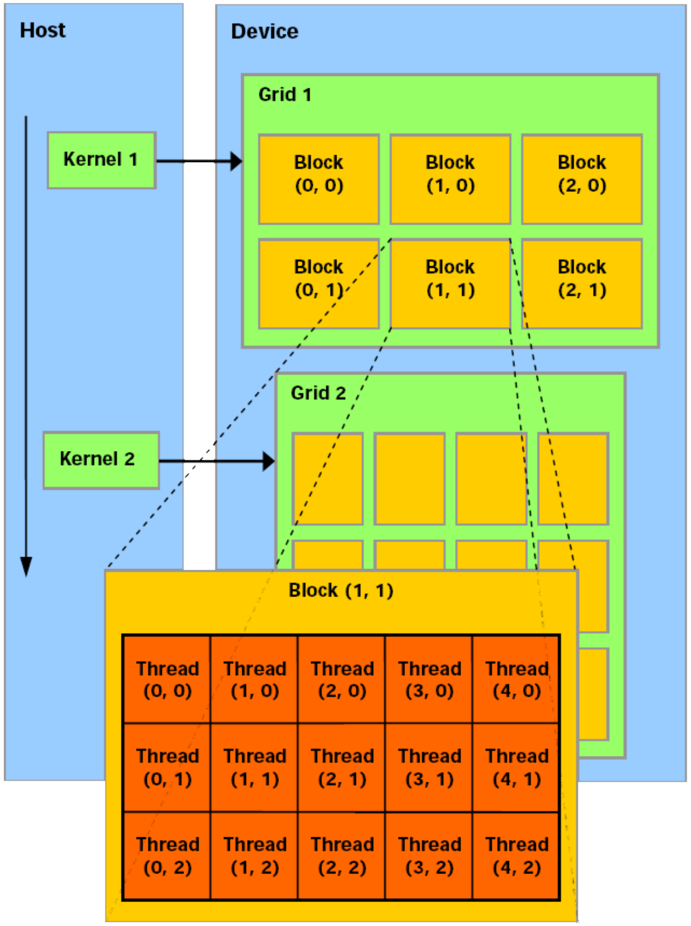
Intel’s many-integrated core (MIC)



	Xeon E5-2670	Xeon Phi 5110P	Tesla K20X
Cores	8	60	14 SMX
Logical Cores	16 (HT)	240 (HT)	2,688 CUDA cores
Frequency	2.60GHz	1.053GHz	735MHz
GFLOPs (double)	333	1,010	1,317
Memory	~16-128GB	8GB	6GB

Emerging hybrid computing infrastructure and technologies

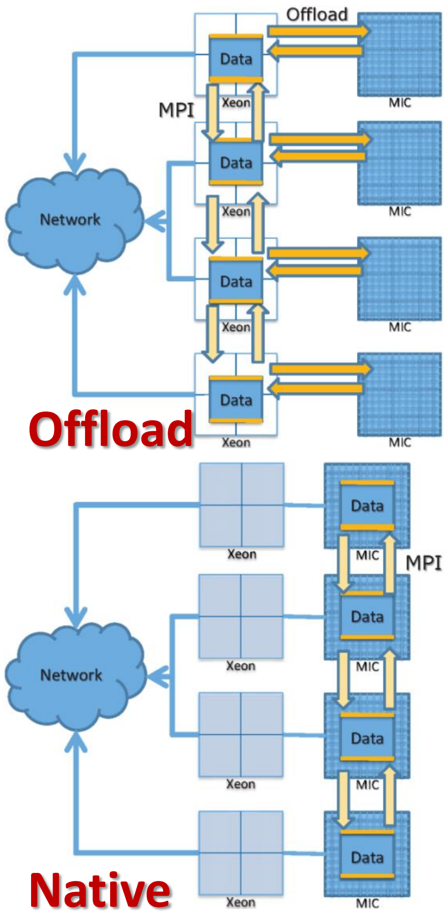
Parallel computing using GPUs



1. Specify the types and sizes of input and output data;
2. Allocate memory on GPU for input data, output data, and intermediate data;
3. Allocate the computing resource on GPU, i.e. specify number of threads per block and total number of blocks;
4. Copy both input and output data from CPU to GPU;
5. Execute the algorithm for data modeling and computation;
6. Copy both input and output data from GPU to CPU;
7. Free the allocated GPU memory.

Emerging hybrid computing infrastructure and technologies

Parallel computing using MICs



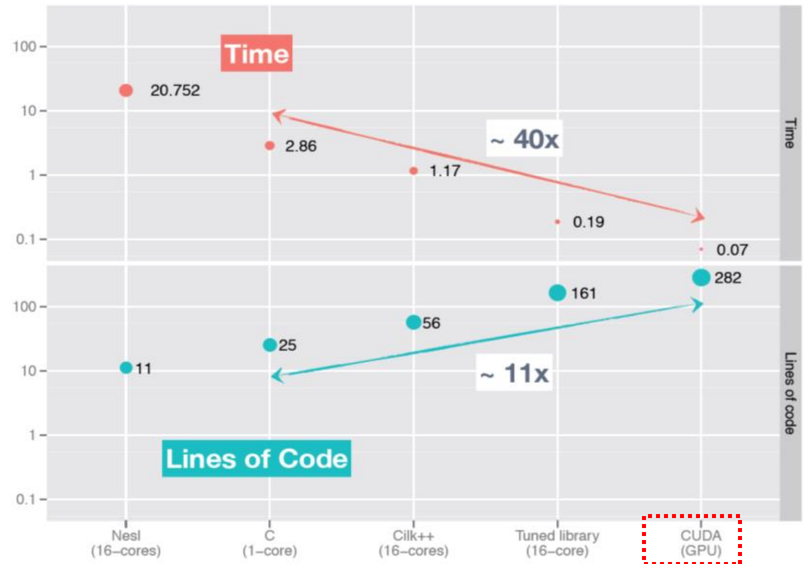
MPI @ CPU + offload
[each MIC card as a 'thread']

MPI @ MIC + OpenMP (multithreading)
[each thread on a MIC core]

MPI @ MIC (Native mode)
[each MIC core as a 'thread']

Significant Algorithmic Re-design and Software Re-engineering

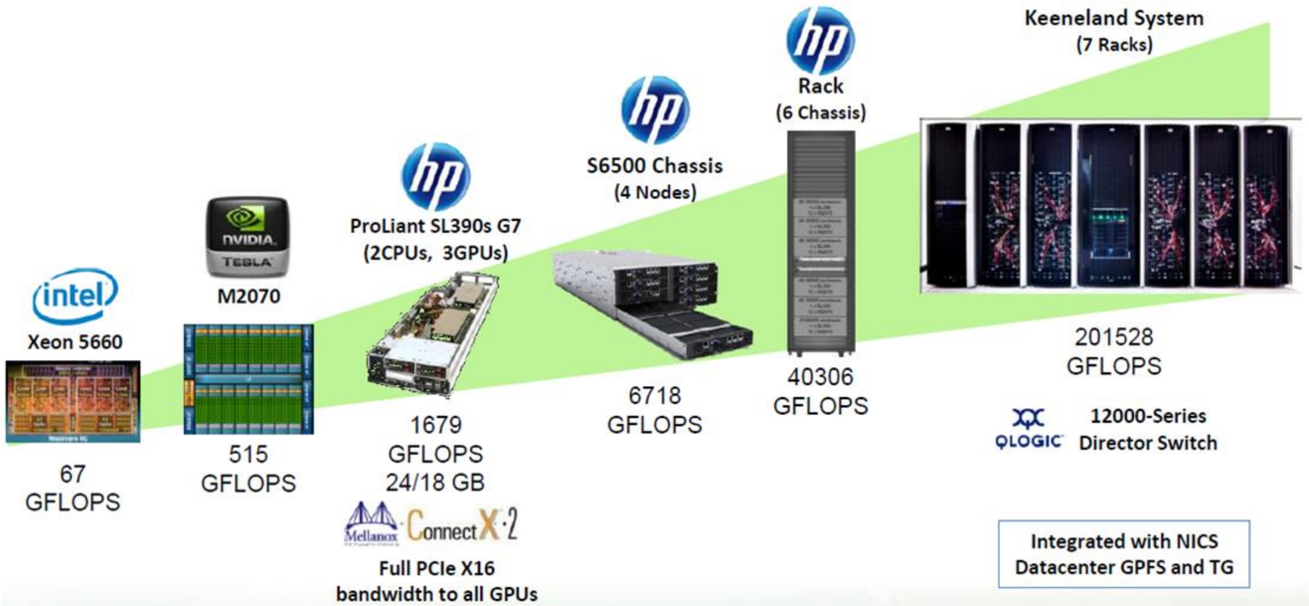
- Software engineering
  - Parallel computing vs. serial program
  - GPU [e.g. CUDA] vs. CPU vs. MIC
  - Communication and control in multiprocessing environment [e.g. MPI, OpenMP]
- Algorithm parallelization and optimization
  - Computational geometry
  - Mathematics
  - Statistics
- Data structure and storage
  - Partitioning
  - Memory
  - I/O



Advanced Heterogeneous Computing Infrastructure

Keeneland

- First hybrid supercomputer sponsored by NSF
- Developed by GA Tech, ORNL, and UTK
- 120 nodes [240 CPUs + 360 GPUs]
- Integrated into XSEDE in 2012, phased out in 2014



We are the first to do geocomputation on Keeneland since 2011

Advanced Heterogeneous Computing Infrastructure

Beacon

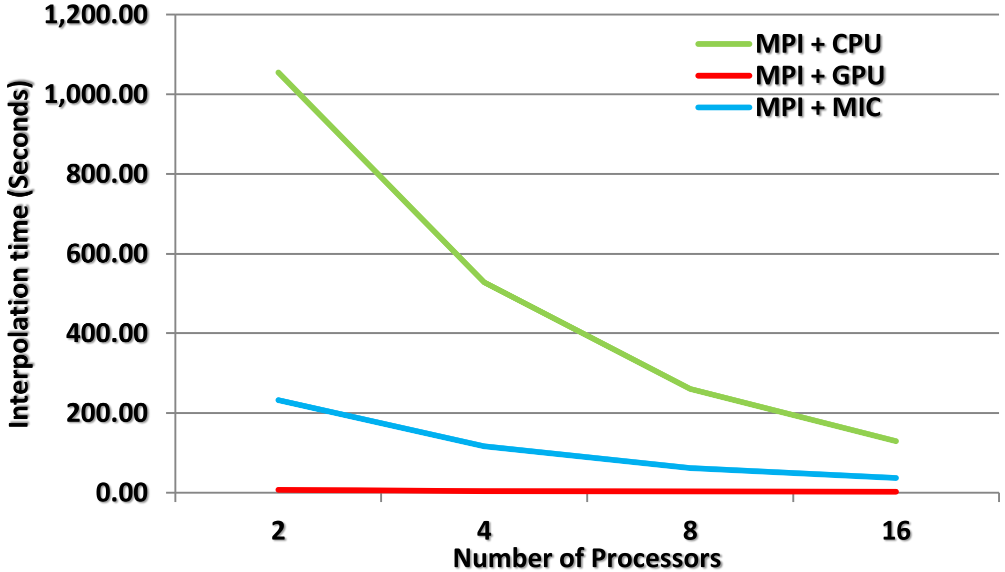
- MIC cluster sponsored by NSF and the State of Tennessee
- Cray CS300-AC Cluster Supercomputer with 48 compute nodes
- Each compute node is equipped with 2 Intel Xeon E5-2670 8-core 2.6 GHz processors + 4 Intel Xeon Phi (MIC) coprocessors 5110P
- Each Xeon Phi coprocessor contains 60 1.053 GHz MIC cores and 8 GB GDDR5 onboard memory.
- Beacon provides 768 conventional cores and 11,520 accelerator cores that provide over 210 TFLOP/s of combined computational performance, 12 TB of system memory, 1.5 TB of coprocessor memory, and over 73 TB of SSD storage

We are the first to do geocomputation on Beacon since 2013

Kriging interpolation on Keeneland vs. Beacon

(no communication between nodes)

Number of processors	Keeneland KIDS								Beacon			
	MPI + CPU				MPI + GPU				MPI + MIC			
	Read	Comp.	Write	Total	Read	Comp.	Write	Total	Read	Comp.	Write	Total
2	0.03	1,054.99	1.13	1,056.15	0.03	6.77	0.90	7.69	1.24	232.43	12.24	245.9
4	0.03	528.43	0.98	529.44	0.03	3.92	0.99	4.93	1.27	116.34	16.44	134.05
8	0.03	260.60	0.95	261.57	0.03	2.76	0.94	3.73	1.23	61.48	54.43	117.14
16	0.03	129.61	1.16	130.69	0.03	2.17	1.11	3.31	1.31	36.74	300.23	338.28



* The processor can be a CPU, a GPU, or Intel MIC in a corresponding implementation.

** The computation time includes both the time spent on data processing and the time spent on communication.

*** Only 360 or 720 MIC cores are used in the computation with 8 or 16 processors, respectively

Iterative Self-Organizing Data Analysis Technique Algorithm (ISODATA) for unsupervised image classification

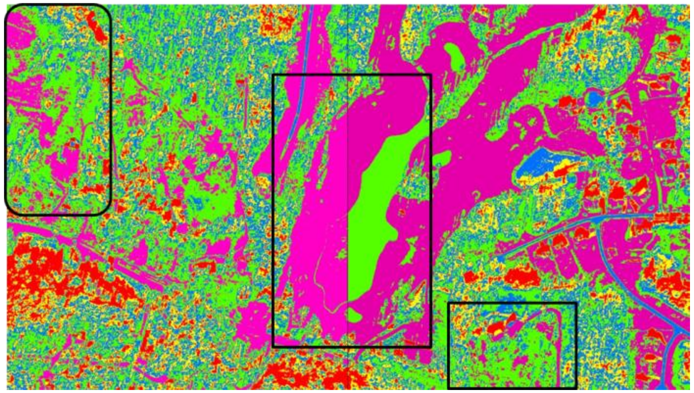
- ISODATA can be implemented in three steps:

1) calculate the initial mean value of each class;
2) classify each pixel to the nearest class; and
3) calculate the new class means based on all pixels in one class.

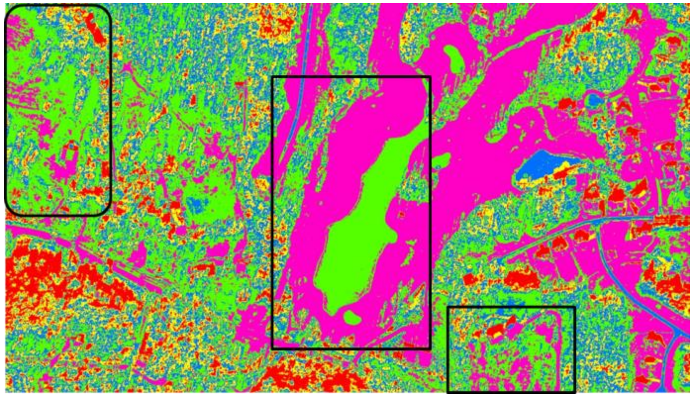
The second and third steps are repeated until the change between two iterations is small enough.

Loose communication needed in each iteration

Symbol	Definition
C	the number of classes to be created
T	convergence threshold which is the maximum percentage (e.g. 95%) of pixels whose class values are allowed to be unchanged between iterations
M	the maximum number of iterations to be performed



Results from two separate processes



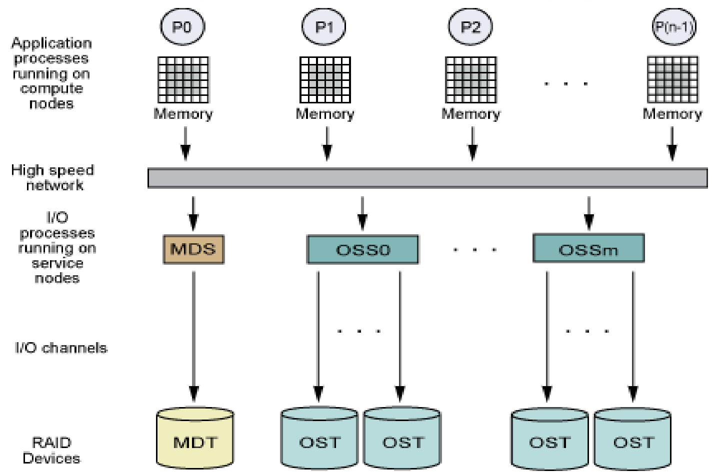
Results from one single process

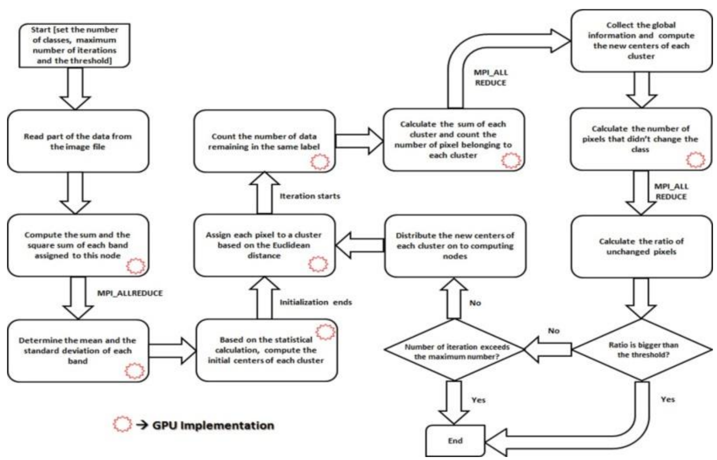
ISODATA on Keeneland vs. Beacon

Performance comparison (time counted by second).
One tile of 18 GB image is classified into 10 classes.
Classification computation can be accomplished in about a half minute when 100 GPUs and 100 MICs are used)

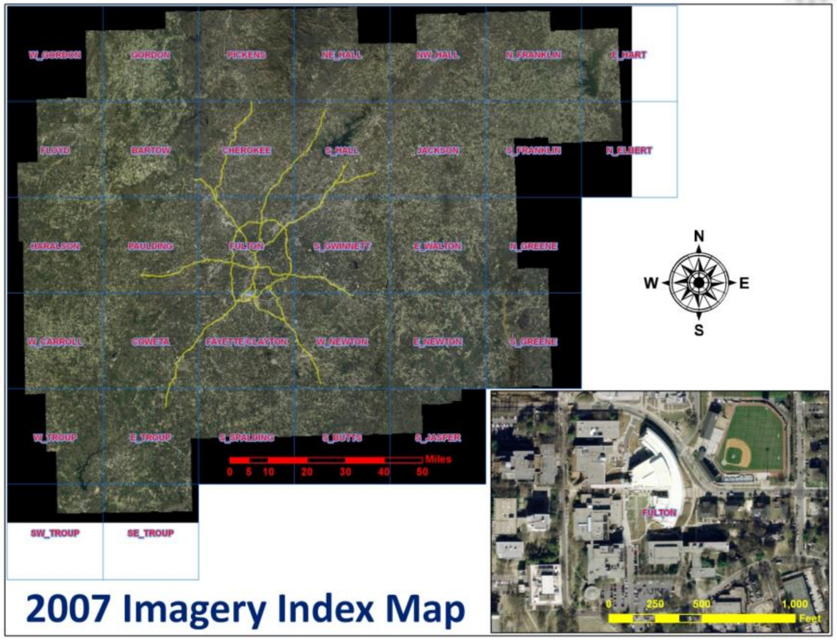
Number of Processors	Keeneland (KIDS)						Beacon		
	MPI + CPU			MPI + GPU			MPI + MIC		
	Read	Comp.	Total	Read	Comp.	Total	Read	Comp.	Total
36	54.59	114.16	168.75	44.90	71.76	116.66	21.88	69.38	91.26
60	44.58	83.95	128.53	43.88	54.45	98.32	874.51	54.03	928.54
64	49.15	65.90	115.05	52.10	46.00	98.10	32.72	39.71	72.43
80	52.50	51.18	103.67	48.97	36.39	85.37	15.37	39.07	54.44
100	1.29	81.31	82.59	36.35	33.81	70.16	41.99	31.52	73.51

Scalability test on supercomputer Kraken using 10,800 cores





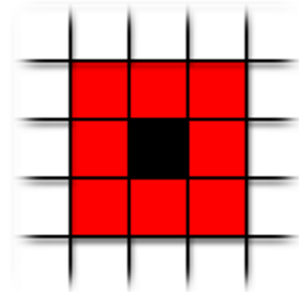
Number of Cores	144	324	576	900
Stripe Count	80	80	80	80
Stripe Size (MB)	10	10	10	10
Read Time (Sec)	5.66	5.13	2.94	2.77
Classification Time (Sec)	13.72	6.15	3.56	3.31



# of tiles	5 classes				10 classes				15 classes				20 classes			
	I/O	CLS	Total	IR	I/O	CLS	Total	IR	I/O	CLS	Total	IR	I/O	CLS	Total	IR
1	4.32	2.13	6.45	4	4.25	8.62	12.87	11	5.51	12.07	17.57	11	6.00	18.13	24.13	13
2	8.94	2.16	11.10	4	20.31	7.92	28.23	10	17.16	11.32	28.47	11	9.02	15.09	24.11	12
4	21.01	2.21	23.23	4	16.40	7.95	24.35	10	14.80	13.41	28.21	13	16.40	7.95	24.35	10
8	28.83	2.23	31.06	4	28.95	7.41	36.36	9	28.67	14.78	43.46	14	29.52	15.34	44.86	12
12	44.86	2.29	47.15	4	45.92	6.57	52.49	8	58.31	9.43	67.74	9	41.56	15.37	56.93	12

Intense communication in spatial modeling by Cellular Automata

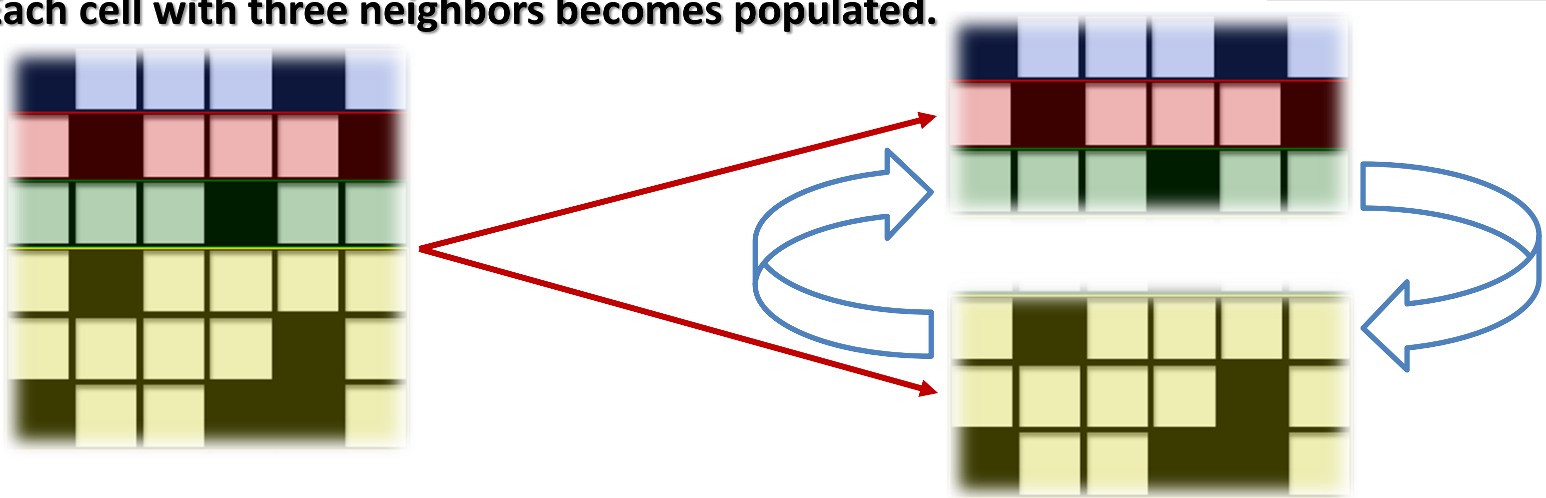
❑ Game of Life (GOL), invented by Cambridge mathematician John Conway, is a well-known generic Cellular Automation (CA) that consists of a collection of cells which, based on a few mathematical rules, can live, die or multiply.



The Rules:

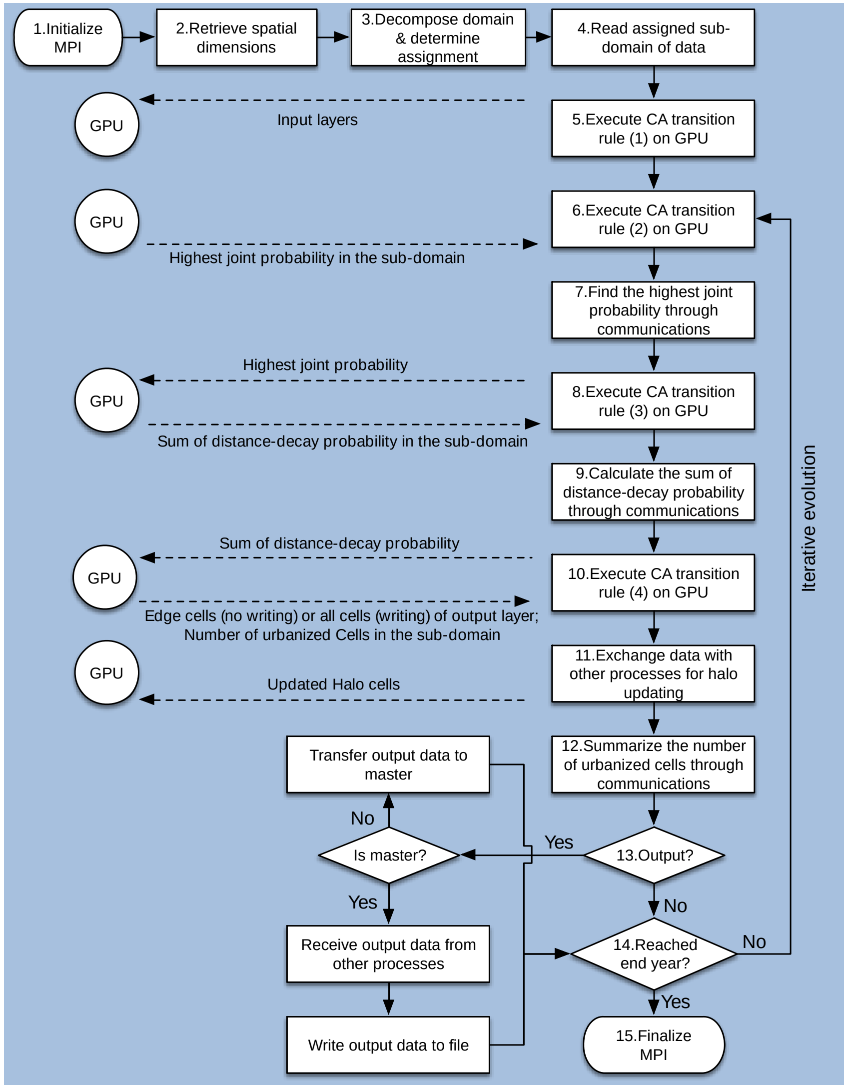
- For a space that is 'populated':
  - Each cell with one or no neighbors dies, as if by loneliness.
  - Each cell with four or more neighbors dies, as if by overpopulation.
  - Each cell with two or three neighbors survives.
- For a space that is 'empty' or 'unpopulated'
  - Each cell with three neighbors becomes populated.

A cell is "born" if it has exactly 3 neighbors, stays alive if it has 2 or 3 living neighbors, and dies otherwise.



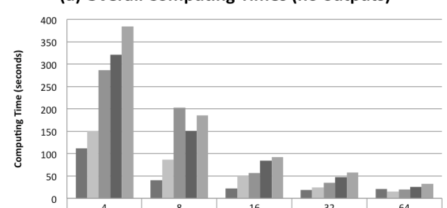
NSF
San Diego State University
Kent State University
University of Arkansas

A hybrid parallel Cellular Automata model for urban growth simulation over GPU/CPU heterogeneous architectures



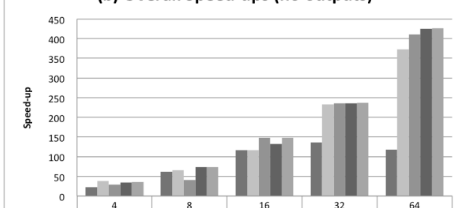
	Reading (s)	Computing (s)	Total (s)
10-year simulation	47.2	2,492.69	2,539.89
20-year simulation	47.26	5,636.26	5,683.52
30-year simulation	46.38	8,283.61	8,329.99
40-year simulation	47.05	11,061.32	11,108.37
50-year simulation	46.97	13,600.04	13,647.01

(a) Overall Computing Times (no outputs)



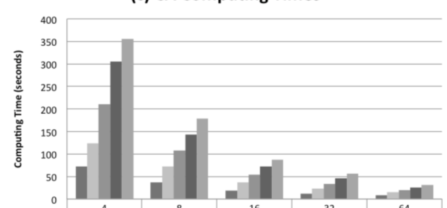
	4	8	16	32	64
10-year	111.47	41.16	21.86	18.65	21.49
20-year	150.9	86.79	48.69	24.45	15.22
30-year	286.6	202.5	56.24	35.38	20.27
40-year	320.7	150.9	84.09	47.1	26.15
50-year	383.38	184.82	91.83	57.76	32.03

(b) Overall Speed-ups (no outputs)



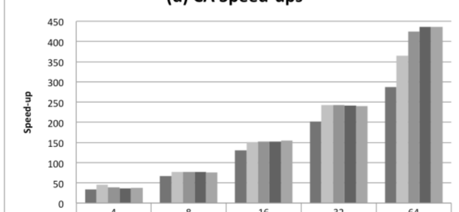
	4	8	16	32	64
10-year	22.79	61.71	116.19	136.19	118.19
20-year	37.66	65.49	116.73	232.45	373.42
30-year	29.06	41.14	148.12	235.44	410.95
40-year	34.64	73.61	132.10	235.85	424.79
50-year	35.60	73.84	148.61	236.27	426.07

(c) CA Computing Times



	4	8	16	32	64
10-year	72.85	37.12	19.13	12.36	8.68
20-year	123.67	72.81	37.43	23.22	15.45
30-year	210.5	108	54.4	34.19	19.5
40-year	304.8	143.4	72.56	45.87	25.38
50-year	355.4	178.5	87.59	56.52	31.21

(d) CA Speed-ups



	4	8	16	32	64
10-year	34.22	67.15	130.30	201.67	287.18
20-year	45.57	77.41	150.58	242.73	364.81
30-year	39.35	76.70	152.27	242.28	424.80
40-year	36.29	77.14	152.44	241.14	435.83
50-year	38.27	76.19	155.27	240.62	435.76

Social media data analytics for understanding the human dynamics in the mobile age

- **Understanding and detecting wildfire events from the chaotic social media through supervised learning**
  - **Focused topic: wildfire events**
  - **Social media: tweets**
  - **Targeted area: California**
  - **Chaos in tweets: pop music, policy discussion, training, budget expense, human relationship, etc.**
  - **True wildfire events: tweet messages about the causes, on-going status, and results of the wildfire**
  - **Exclusion: tweets without the keyword of “wildfire” or “fire”**

Social media data

- 5 datasets downloaded from SDSU's smart dashboard of this project:

Total number of tweets	Tweets with the keyword	Time period
5,355	4,391	07/18/2014 to 10/01/2014
3,340	3,167	05/09/2014 to 06/12/2014
1,888	1,529	01/01/2015 to 05/06/2015
583	408	05/01/2015 to 06/30/2015
139,591	129,743	07/01/2015 to 07/20/2015

Data preprocessing and transformation

- Text retrieving
  - Based on key words of interest to this project
- Text cleaning
  - Uncertain patterns or terminologies in social media [e.g. abbreviated terms, jargons, etc.]
- String matching
  - Lots of algorithms can be applied
- Transformation and localization
  - Key words replacement and standardization
  - Vocabulary unique in California

Data mining and machine learning

- **High frequency word construction**
  - Time series of wildfire events in social media and network
  - Identification of duplicated/retweeted tweets
- **Supervised classification**
  - Training data development and validation
  - Similarity and difference
  - Exemplar identification through Affinity Propagation (AP)
  - Support Vector Machine (SVM) for classification

Initial results

- The new data has about 140K tweets. Tweets without the key word of "wildfire" or "fire" were then removed. The remaining number of tweets is about 130K;
- Within this 130K data, 61,122 of them were at least repeated/retweeted twice. Thus this subset of data is used for testing and validation. The other part of the data, which contains those appeared only once, was processed but the classification results are not validated yet because we don't have sufficient man power to go through 60+K tweets;
- Within 61,122 tweets, however, 8,978 were talking about wildfire events happened in Canada, Washington State, and Oregon State, thus were removed to focus on the events in CA [as local attributes in CA are applied in this study];

Initial results

- Within the remained 52,144 tweets, three approaches were applied to justify whether a tweet message is about a true wildfire event or not:
  - The first approach directly apply the SVM classifier constructed based on the prior four datasets. 24,047 tweets were classified as "true" [group 1], 28,097 tweets were classified as "not true" [group 0]. We manually checked result and found a total of 6,140 tweets was not correctly classified in group 1. Thus the accuracy ratio is about 78% in group 1;
  - The second approach first identified the exemplars in the new data, manually match the exemplars to the correct class, and then applied SVM to do classification [exemplars were excluded]. We manually checked result and found a total of 739 tweets was not correctly classified in group 1. Thus the accuracy ratio is about 97% in group 1;
  - The third approach first identified the exemplars in the new data, automatically match the exemplars to the correct class through similarity calculation and AP cluster analysis, and then applied SVM to do classification [exemplars were excluded]. We manually checked result and found a total of 1220 tweets was not correctly classified in group 1. Thus the accuracy ratio is about 95% in group 1

Initial results

- **The results for group 0 [tweets not about true wildfire events] would generated a lower ratio of correctness [78%, 86%, and 80% in accordance to the three approaches mentioned above].**
  - **This is because *tweets in group 0 have higher noisy problems*, thus even when some tweets would really talk about the true wildfire events, the vocabulary used in the messages could be much closer/similar to those noisy messages.**
- **In conclusion, the 3rd approach could be valid and more effective, although the 2nd approach may generate higher correct ratio**
  - **When human beings are involved, this approach may lead to labor intensive tasks when larger volume of tweets are to be processed. The 3rd approach is thus more significant and practical in this study.**

Current and future initiatives

- **When large volume of data has to be processed, parallel and distributed computing solutions have to be developed, such as GPUs and MICs**
  - **In the case of AP, for example, 10K points need 4GB memory, 20K points need 16GB memory, and 40K points need 64GB memory**
  - **The time used for training SVM classifiers grows when the size of the training data increases.**
    - **The theoretical computation complexity of building a SVM classifier lies between $O(n^2)$ to $O(n^3)$**
  - **In the case of ABM, data communication may have to be executed multiple times in order to complete the computational processes thus increase the difficulty and challenge in development.**
  - **Other tasks of big data processing and analytics in this project**
    - **Vaccine, political debate, election, etc.**

Thank you

Questions ?

Suggestions ?