Status of Taiwan Group

Guo Chin Liu
Tamkang University

F2F meeting @ Toyama 2018

TGWG conference (2018)

- Time: Oct. 7(Sun.)-10(Wed.)
- Venue: Tamkang University in New Taipei City
- Invited Speaker:
  - Lior Burko on Strong gravity
  - Kipp Cannon on Data analysis
  - Yan-Bei Chen on Quantum measurements or gravity theory related
  - Qing-Guo Huang on Primordial black holes
  - Tjonnie Li on Parameter estimations
  - Amy Lien on Multi-Messenger Astronomy
  - Rafael Porto on EFT of Post-Newtonian (To be Confirmed)
  - Jan Steinhoff On Gravitational waveform construction
  - Barry Wardell on self-force

Summary

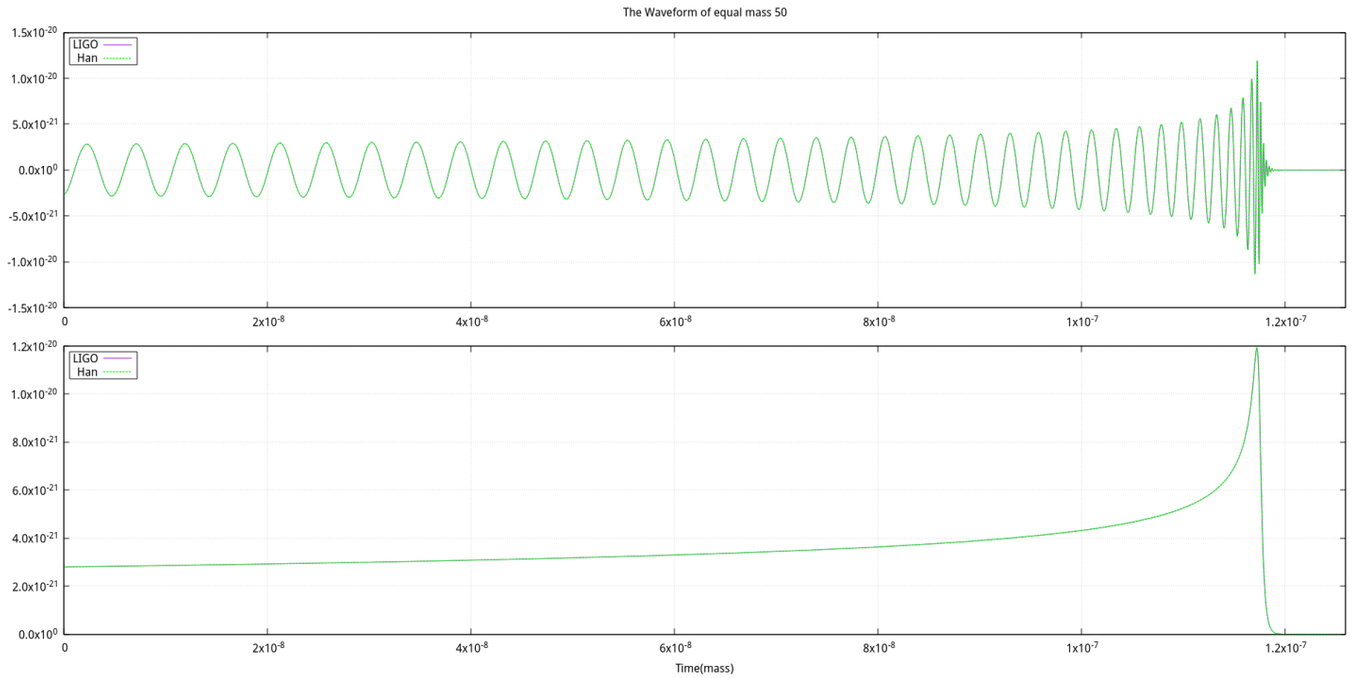
- EOB waveform generation (Kagali)
- CBC search with GPU
- Deep Learning

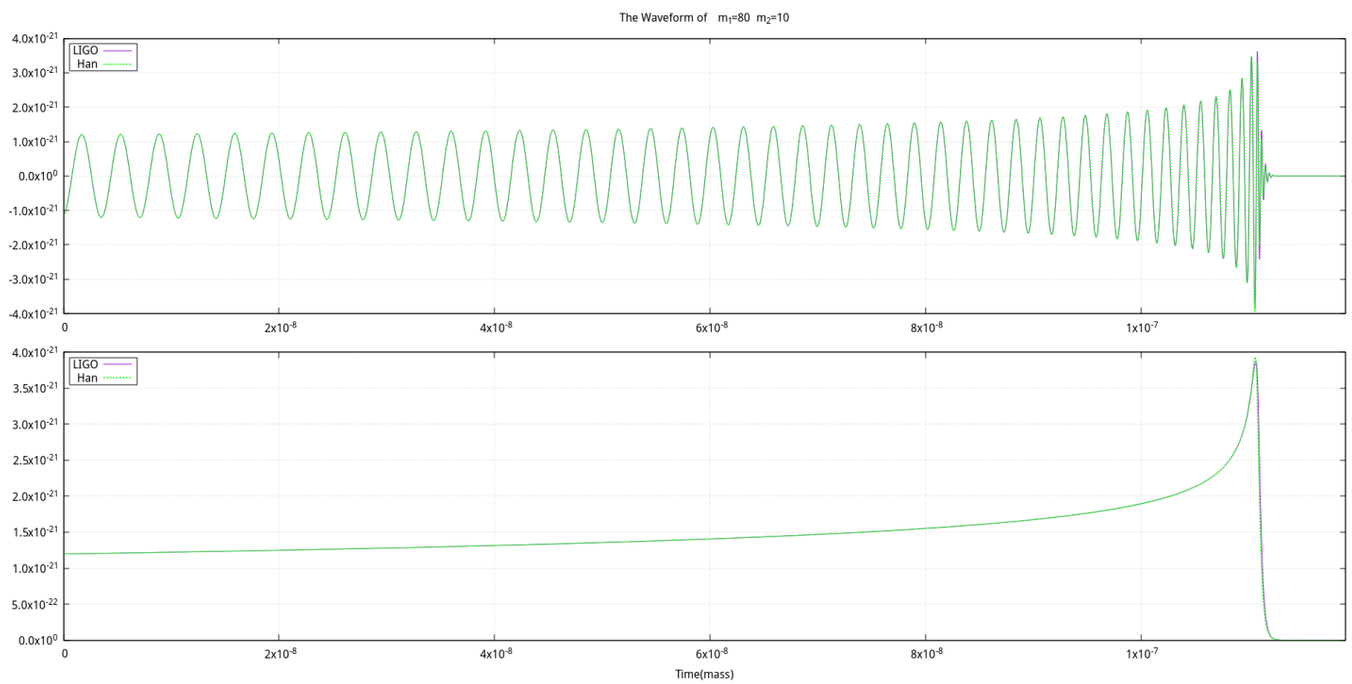
EOB Waveform Generation

Guo-Zhang Huang (M. Student)

- SEOBNRE model: Spin+EOB+NR+Eccentricity (Cao & Han 2017)
  - dynamics of two bodies, presented by Hamiltonian
  - GW emitted by binary system (quasicircular + eccentric)
  - radiation-reaction force

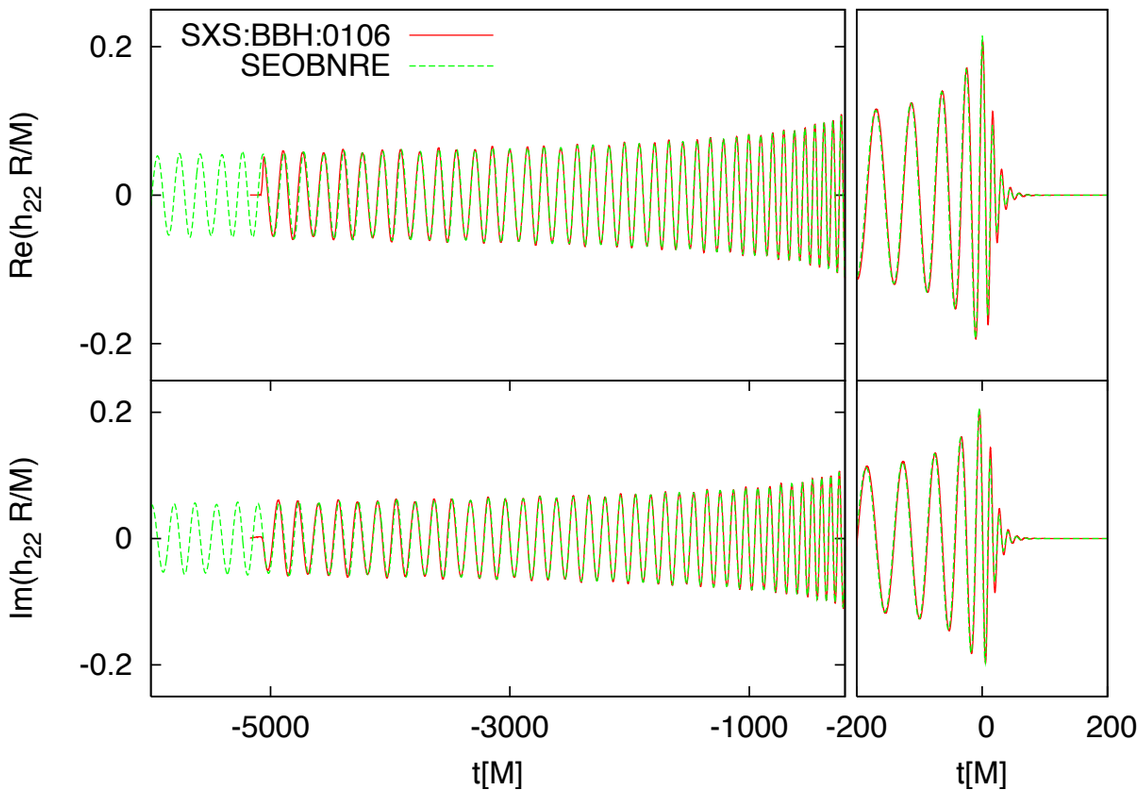
Consistency of SEOBNRv1 and SEOBNRE



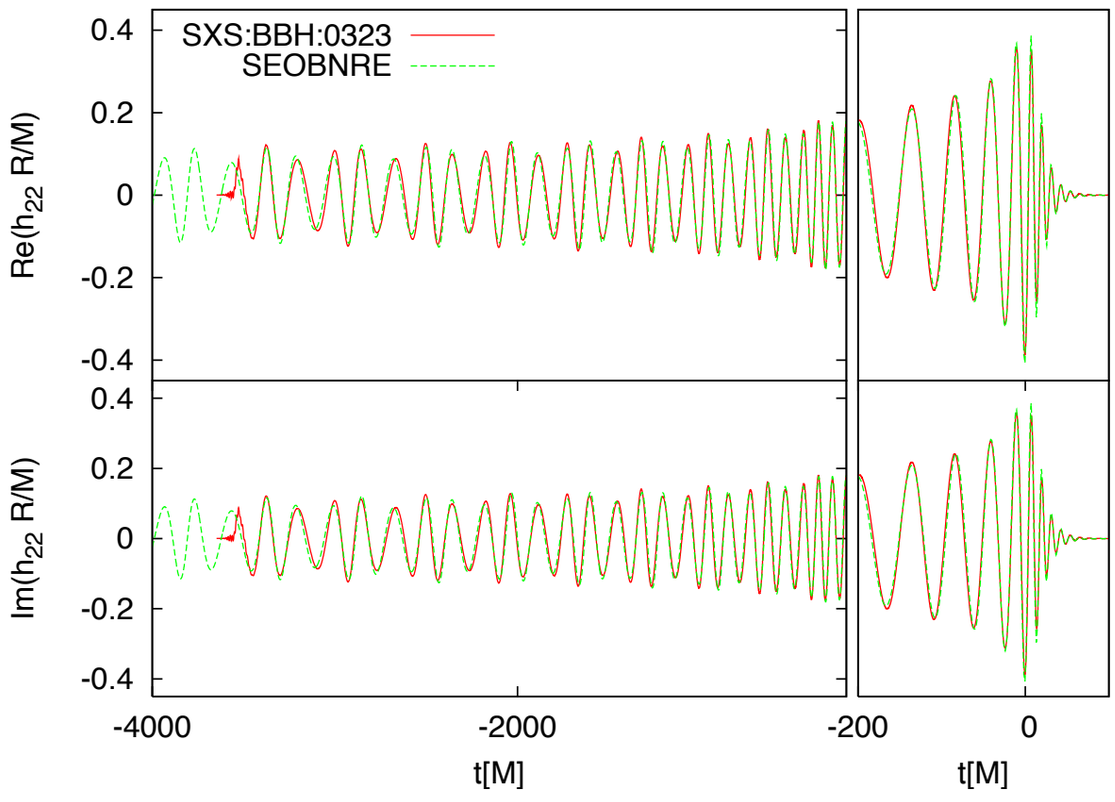


Comparison with SPEC Simulation

e_0=0.02181 for Mf_0=0.00147764
q=5
spinless



e_0=0.3 for Mf_0=0.00147764
q=11/9
x_1=0.33, x_2=-0.44



Cao & Han 2017

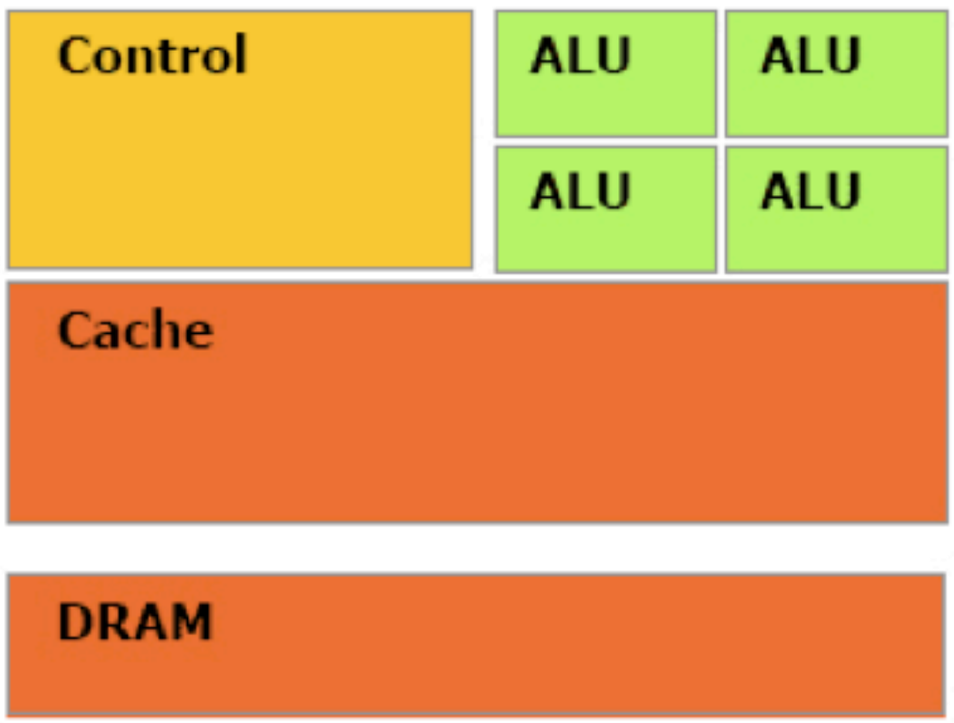
GPU acceleration

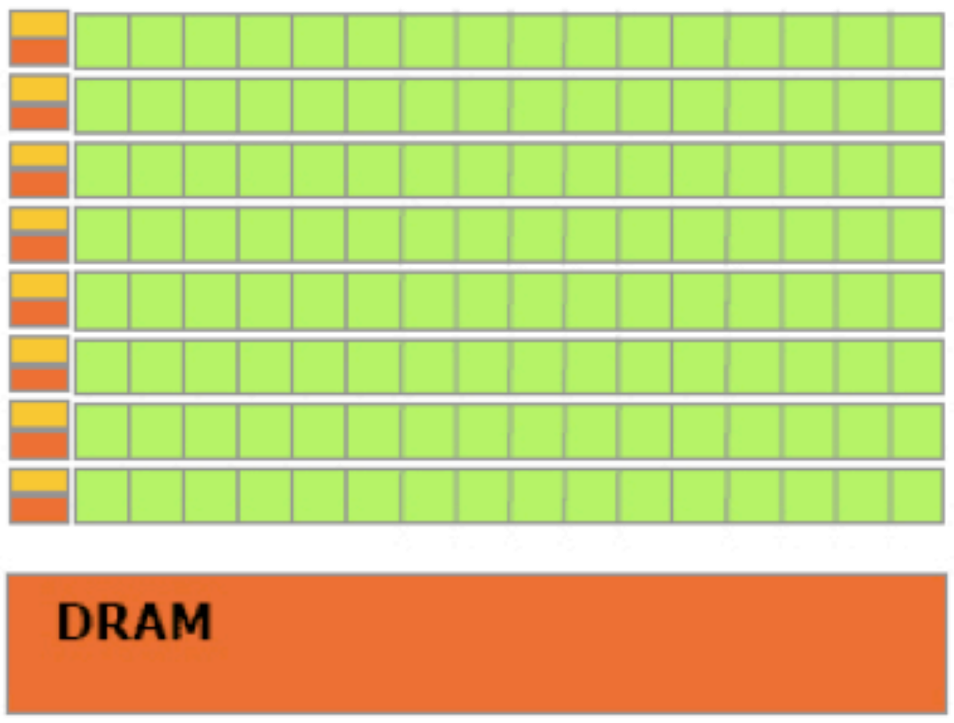
Han-Shiang Kuo (M. Student)

- Motivation:
  - learning search algorithm, but lack of computing power
  - not only for the search, but maybe applied to the other some packages in the softwares (e.g. Haino san's parameter estimation)

Advantages on GPU

- High throughput oriented
- Compute cores occupy most part of die
- Good for Linear Algebra operations, FFT, etc





Nvidia web.

Specs of Selected GPUs

	Double precision	Single precision	Memory	Memory Bandwidt	Cores
GTX1080	0.35Tflops	8.87Tflops	8GB	320GB/s	2560
K80	2.91Tflops	8.73Tflops	24GB	480GB/s	4992
Titan X	0.31Tflops	10.9Tflops	12GB	480GB/s	3072
P100	4.7Tflops	9.3Tflops	16GB 12GB	732GB/s 549GB/s	3584

Flops: Floating-point operations per seconds

start

Read Strain data

apply high-pass filter

FFT

Estimate PSD

Read PSD estimated from long data

generate template bank

Loop for each template bank

Matched-filter

$\rho > \rho^*$

No

yes

Flowchart by Tagoshi-san and Yuzurihara-san
Usman et al. 2016
Allen et al. 2012

Calculate boundary for chi-square veto

Clustering MF and chi-square

Calculate detection statistics

exceed threshold

No

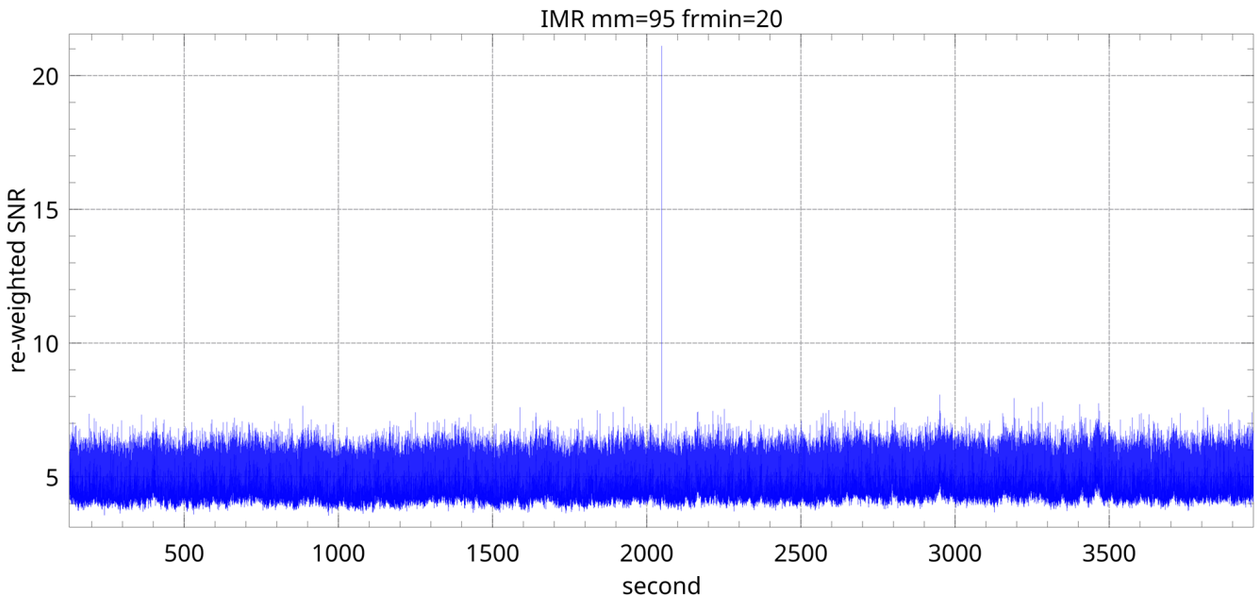
yes

add candidate event to trigger list

End of Loop

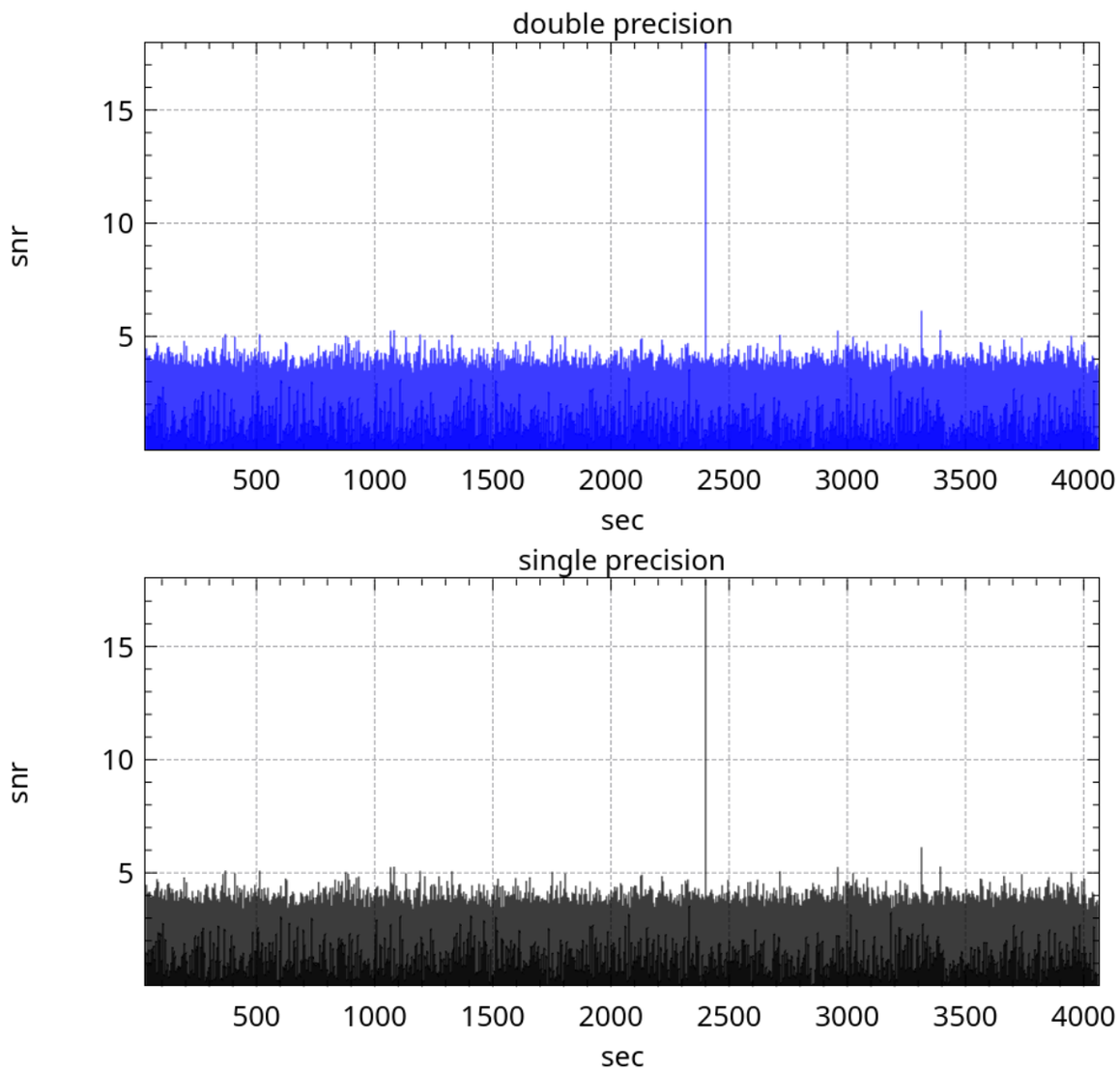
Speed-up Results

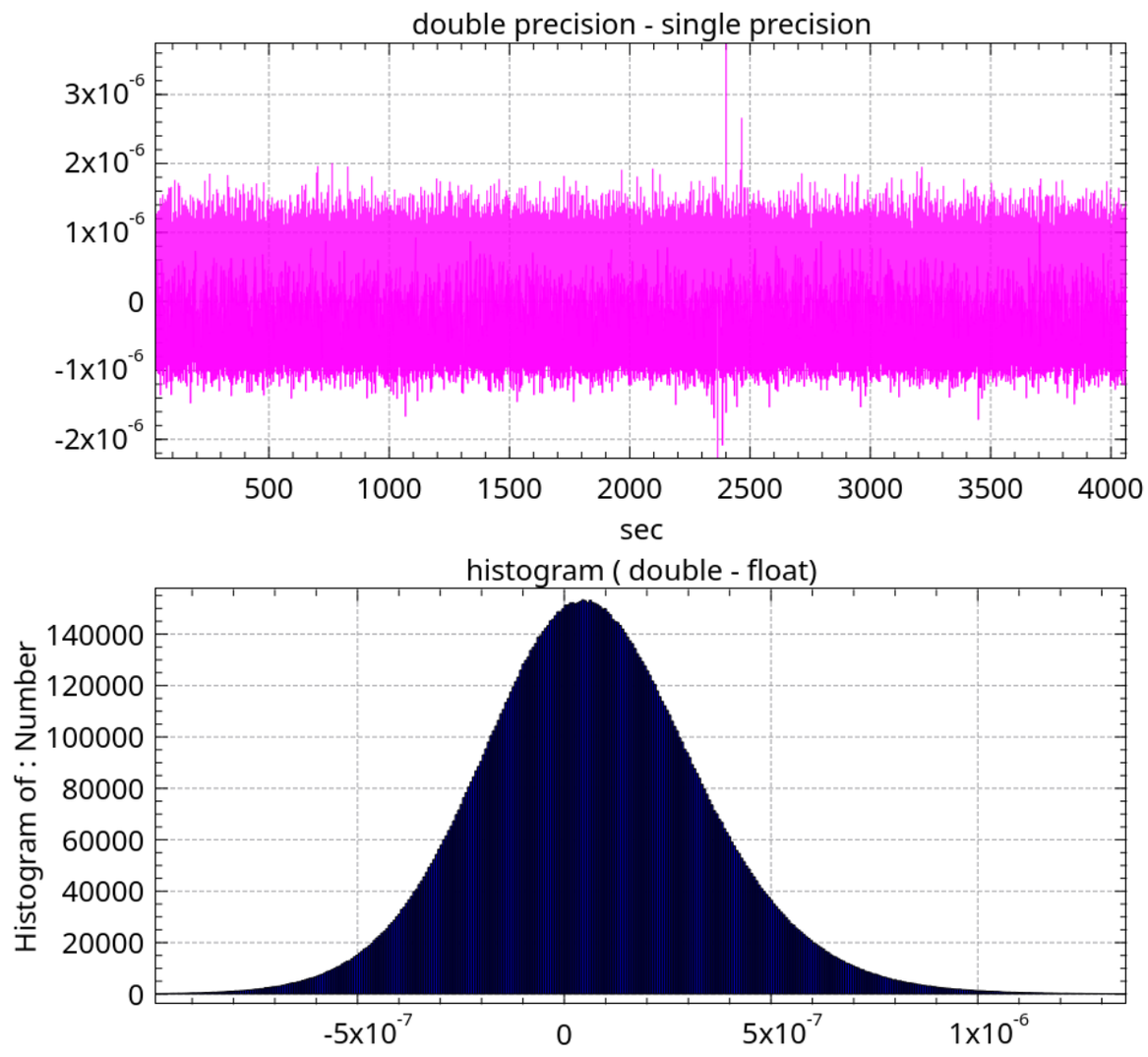
- P100 GPU
- LIGO data with 4096 sec.
- Chunk size =512 secs with 1/2 overlap.
- 110000 IMR templates
- Reading data & PSD estimation ~10 sec
- Template generation ~ 150 mins
- Matched-filter + chi2 veto~ 50 mins
- About 300 times speed up comparing with one cpu core (E2630v2)




```
cufftHandle plan;
cufftPlan1d( &plan, nt, CUFFT_Z2D, batch);
cufftExecZ2D( plan, dfdata, dtdata);
cufftDestroy( plan);
```

Single Precision Test @K80

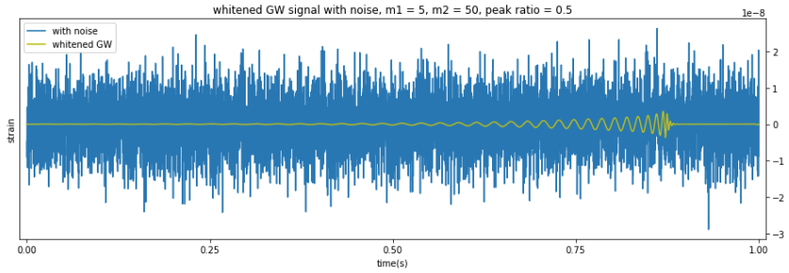




4 times faster for single-precision than double-precision at K80

Deep Learning

Wei-Ren Xu(M. student)
Jin-Yau Tang (U)
Yukari Uchibori (U)



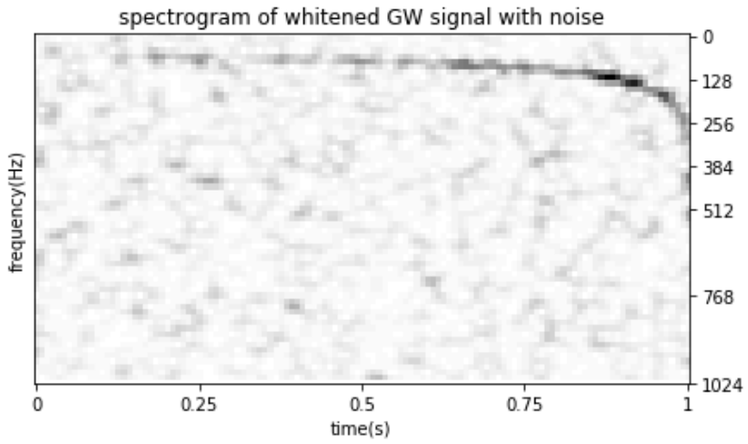
- Training part:
  - 1 second data with 8192 sampling rate
  - whiten Signal(by EOBNRv2) + white noise 9462
  - Pure white noise 8000
  - Mass range: 5-75 Ms in steps of 0.5Ms, mass ratio<10
- Testing part:
  - 3492 test data (S+N/N)
  - Running time about a few seconds (accelerated by K80)

	Input	vector (size: 8192)
1	Reshape	matrix (size: 1 × 8192)
2	Convolution	matrix (size: 64 × 8177)
3	Pooling	matrix (size: 64 × 2044)
4	ReLU	matrix (size: 64 × 2044)
5	Convolution	matrix (size: 128 × 2014)
6	Pooling	matrix (size: 128 × 503)
7	ReLU	matrix (size: 128 × 503)
8	Convolution	matrix (size: 256 × 473)
9	Pooling	matrix (size: 256 × 118)
10	ReLU	matrix (size: 256 × 118)
11	Convolution	matrix (size: 512 × 56)
12	Pooling	matrix (size: 512 × 14)
13	ReLU	matrix (size: 512 × 14)
14	Flatten	vector (size: 7168)
15	Linear Layer	vector (size: 128)
16	ReLU	vector (size: 128)
17	Linear Layer	vector (size: 64)
18	ReLU	vector (size: 64)
19	Linear Layer	vector (size: 2)
	Output	vector (size: 2)

Architecture of deep neural network (George & Huerta 2018)

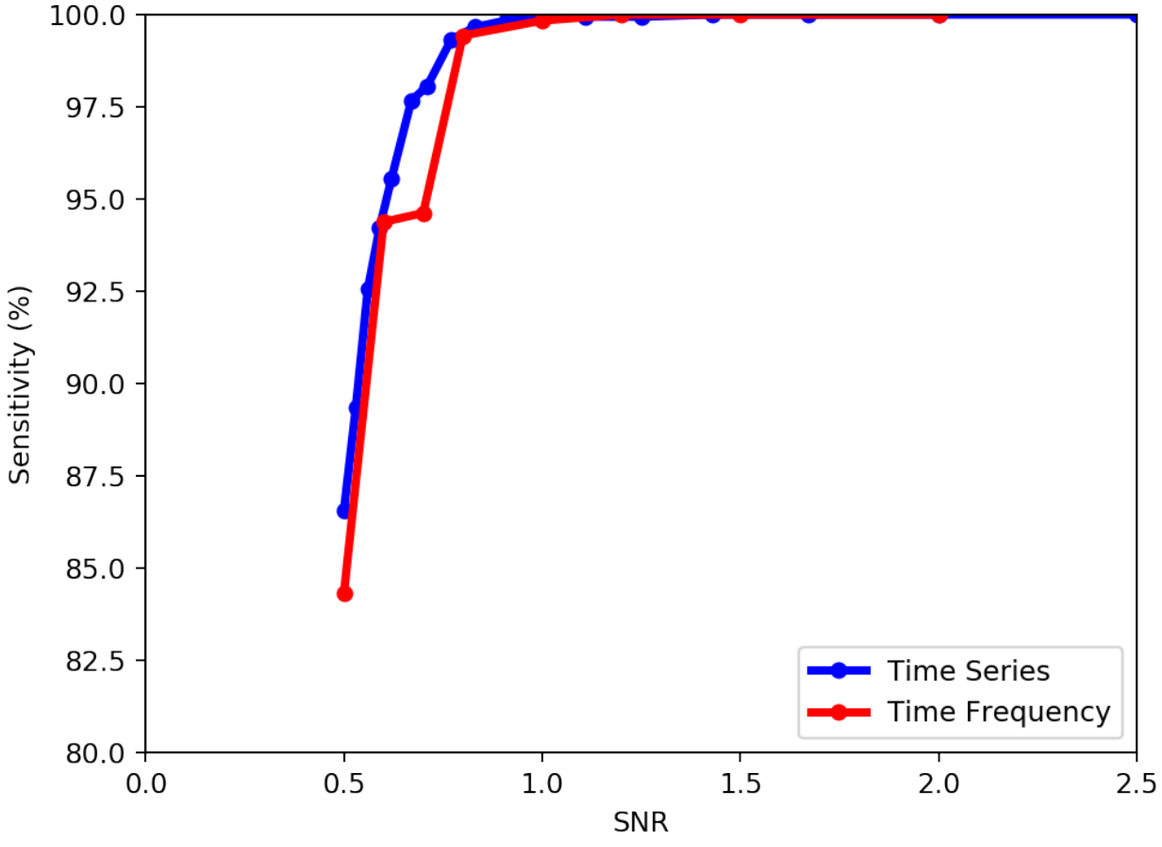
Deep Learning

- Training part:
  - 1 second data with 8192 sampling rate
  - whiten Signal(by EOBNRv2) + white noise 9462
  - Pure white noise 8000
  - Mass range: 5-75 Ms in steps of 0.5Ms, mass ratio<10
- Testing part:
  - 3492 test data (S+N/N)

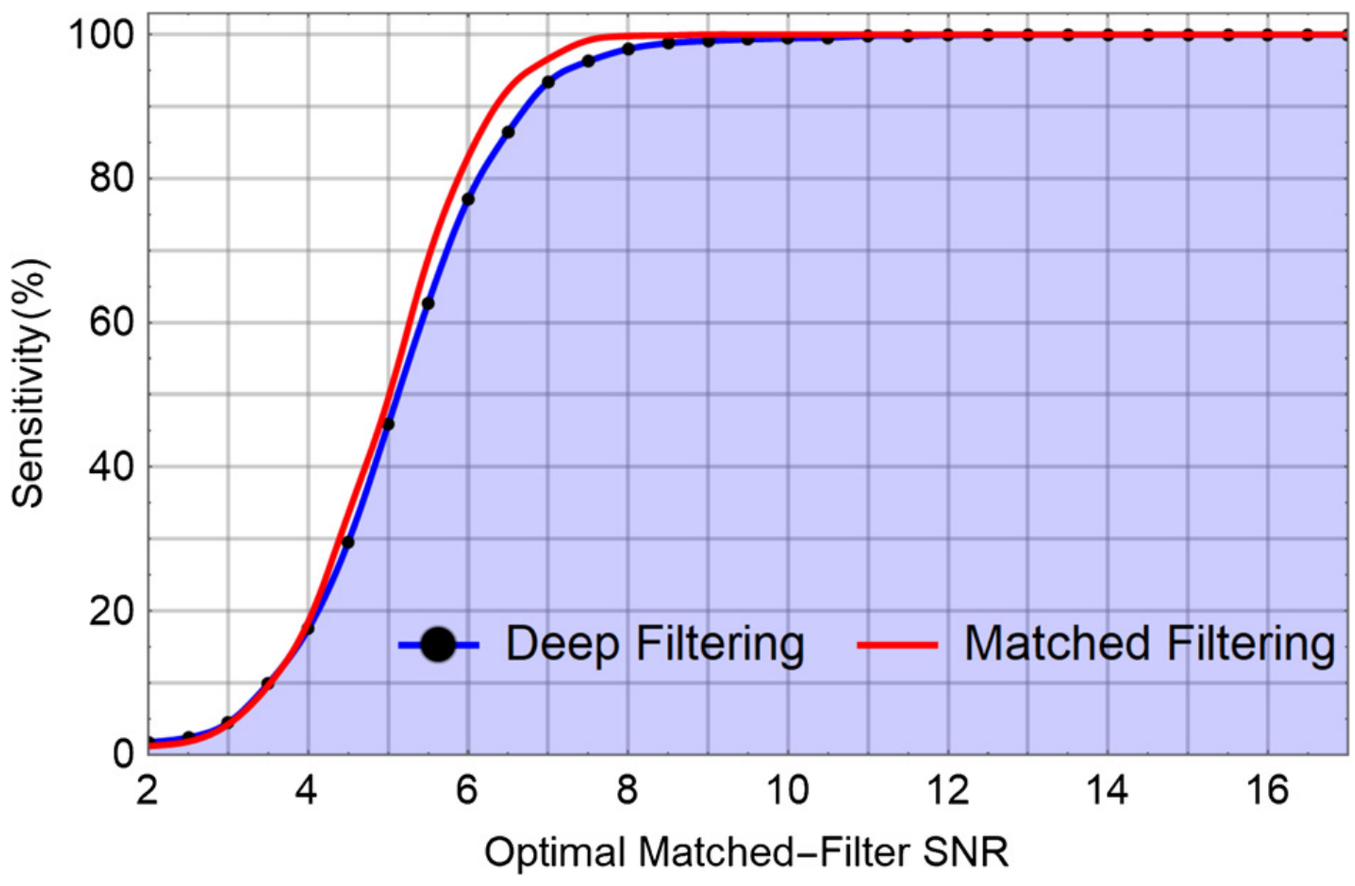


1	Input	Matrix (size: [33,121])
2	Reshape	Matrix (size: [33,121,1])
3	Maxpooling	Matrix (size: [17,61,1])
4	Convolution	Matrix (size: [17,61,4])
5	ReLU	Matrix (size: [17,61,4])
6	Convolution	Matrix (size: [17,61,4])
7	Shortcut Connection	Matrix (size: [17,61,4])
8	ReLU	Matrix (size: [17,61,4])
9	Maxpooling	Matrix (size: [9,31,4])
10	Convolution	Matrix (size: [9,31,5])
11	ReLU	Matrix (size: [9,31,5])
12	Convolution	Matrix (size: [9,31,5])
13	Shortcut Connection	Matrix (size: [9,31,5])
14	ReLU	Matrix (size: [9,31,5])
15	Maxpooling	Matrix (size: [5,16,5])
16	Convolution	Matrix (size: [5,16,6])
17	ReLU	Matrix (size: [5,16,6])
18	Convolution	Matrix (size: [5,16,6])
19	Shortcut Connection	Matrix (size: [5,16,6])
20	ReLU	Matrix (size: [5,16,6])
21	Flatten	Vector (size: [480,])
22	Linear Layer	Vector (size: [128,])
23	ReLU	Vector (size: [128,])
24	Linear Layer	Vector (size: [32,])
25	ReLU	Vector (size: [32,])
26	Linear Layer	Vector (size: [2,])
27	Output(SoftMax)	Vector (size: [2,])

Results for Detection



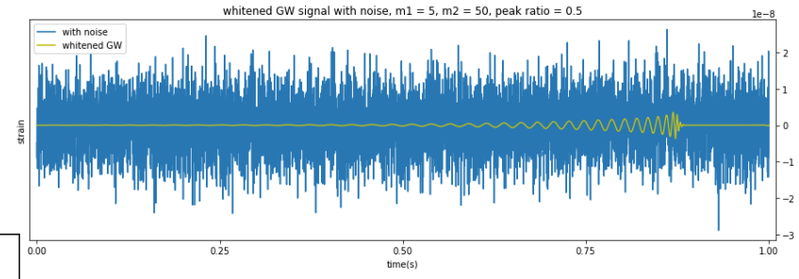
snr=max(h)/(sd of noise)



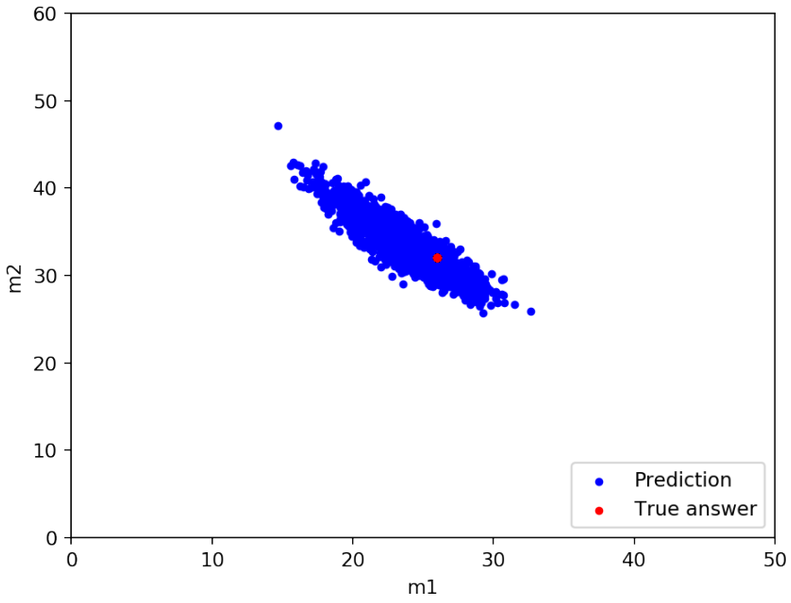
Results for Mass Estimation

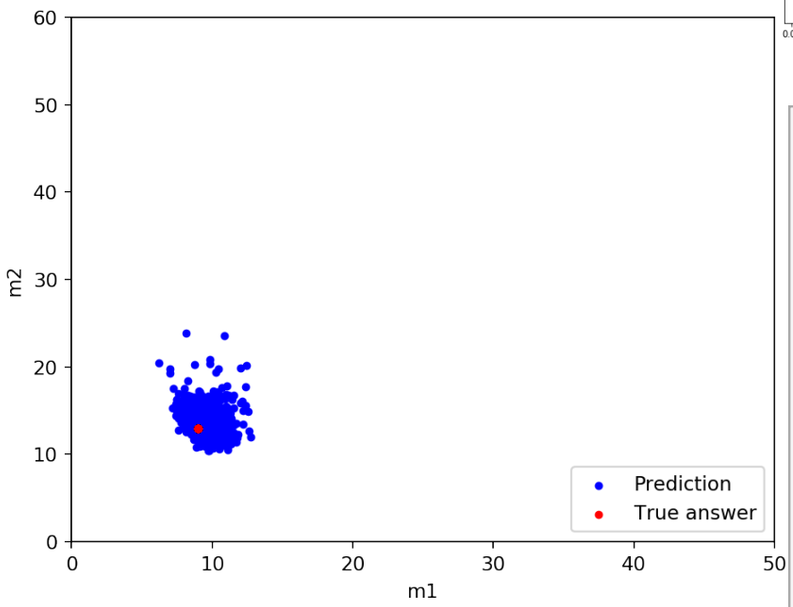
input (32,26)

input (9,13)

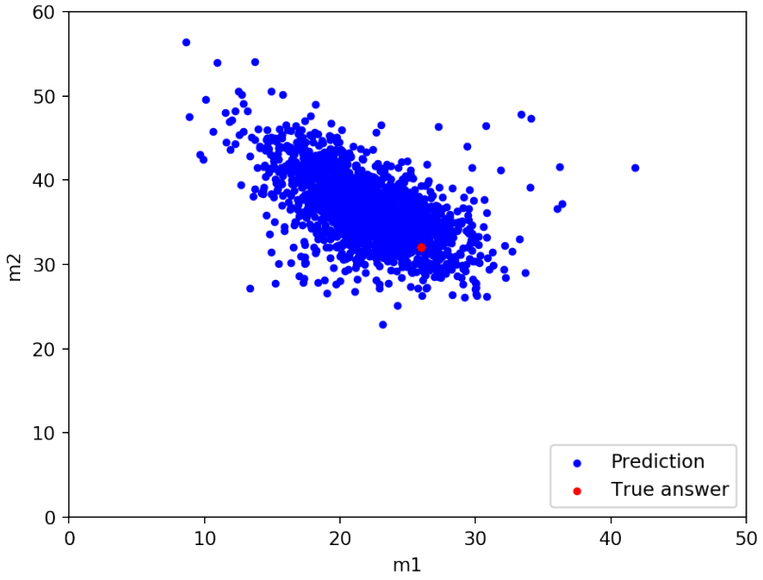


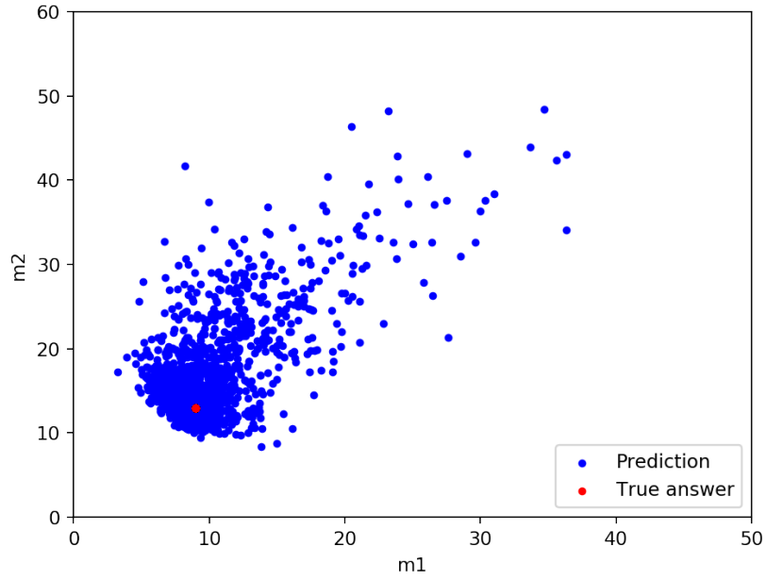
SNR=2





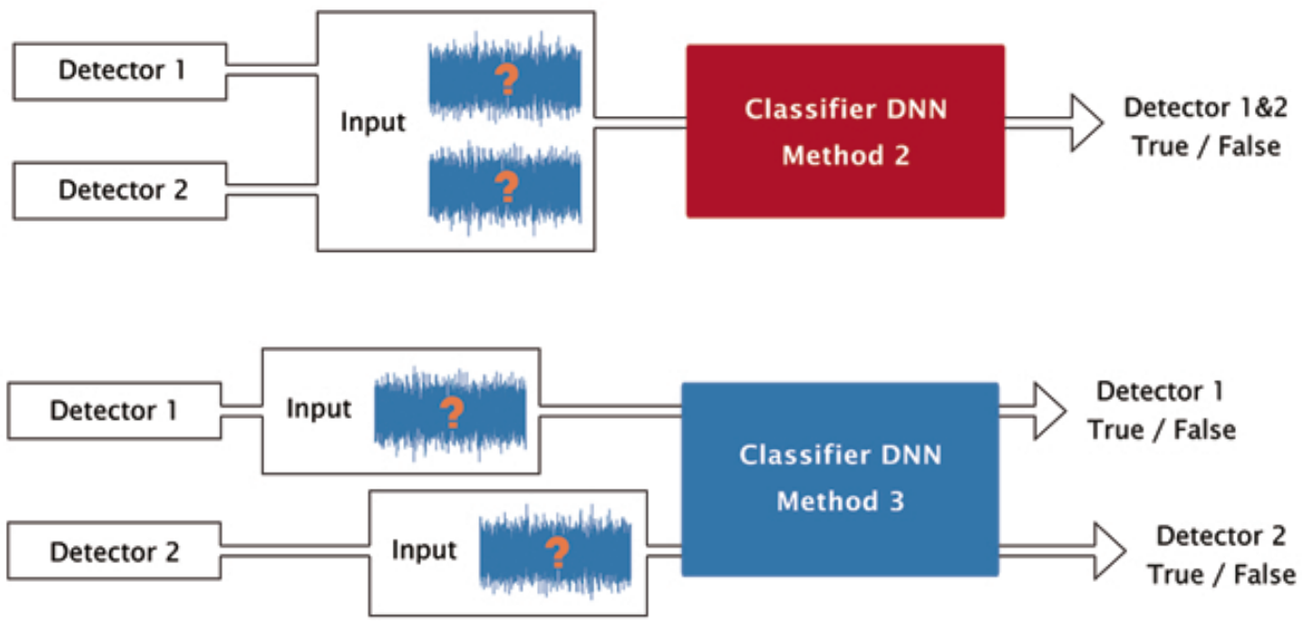
SNR=0.83

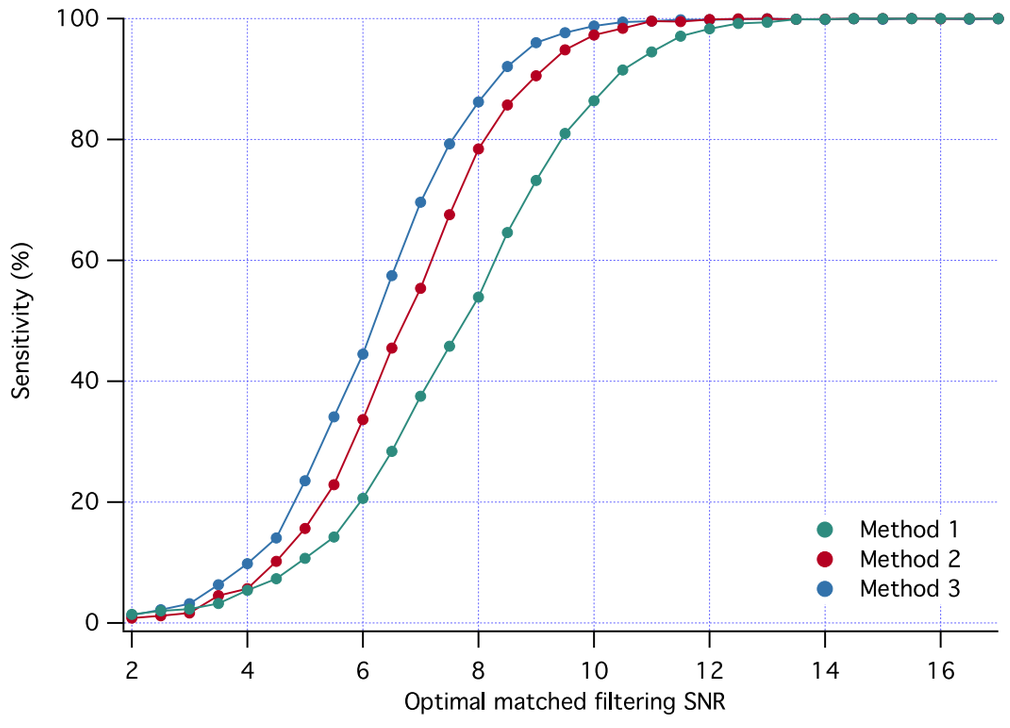




	Input	vector (size: 8192)
1	Reshape	matrix (size: 1 × 8192)
2	Convolution	matrix (size: 64 × 8177)
3	Pooling	matrix (size: 64 × 2044)
4	ReLU	matrix (size: 64 × 2044)
5	Convolution	matrix (size: 128 × 2014)
6	Pooling	matrix (size: 128 × 503)
7	ReLU	matrix (size: 128 × 503)
8	Convolution	matrix (size: 256 × 473)
9	Pooling	matrix (size: 256 × 118)
10	ReLU	matrix (size: 256 × 118)
11	Convolution	matrix (size: 512 × 56)
12	Pooling	matrix (size: 512 × 14)
13	ReLU	matrix (size: 512 × 14)
14	Flatten	vector (size: 7168)
15	Linear Layer	vector (size: 128)
16	ReLU	vector (size: 128)
17	Linear Layer	vector (size: 64)
18	ReLU	vector (size: 64)
19	Linear Layer	vector (size: 2)
	Output	vector (size: 2)

Multi-Detector





Computing Power

- budget for renting to next July: 500KNTD, and 250KNTD for next year
- 1.68 NTD/1GPU-hr, and 0.07NTD/1CPU-hr (0.3M GPU-hr or 7M cpu-hr till next July)
- Typically they are shared cores, but 3 nodes with GPU are dedicated in TGWG
- Largest queue is 2000 cores.

Node Type	Node range	Total units (nodes)	Compute resources per unit (node)					
			CPU Sockets	CPU cores	Memory (GB)	Tesla P100	10Gbps interface	480 GB SSD
Thin nodes	cn0101 – cn0673	438	2	40	192	-	-	-
Thin nodes	cn0701 – cn0764	64	2	40	192	-	1	-
Fat nodes	cn0801 – cn0864	64	2	40	384	-	-	-
Fat nodes	cn0901 – cn0964	64	2	40	384	-	-	1
GPU nodes	cn1001 – cn1064	64	2	40	192	4	-	-
Big memory node	cnbm01	1	4	96	6000	-	-	-

summary of NCHC Taiwanian (from NCHC web.)