중력파 데이터 분석

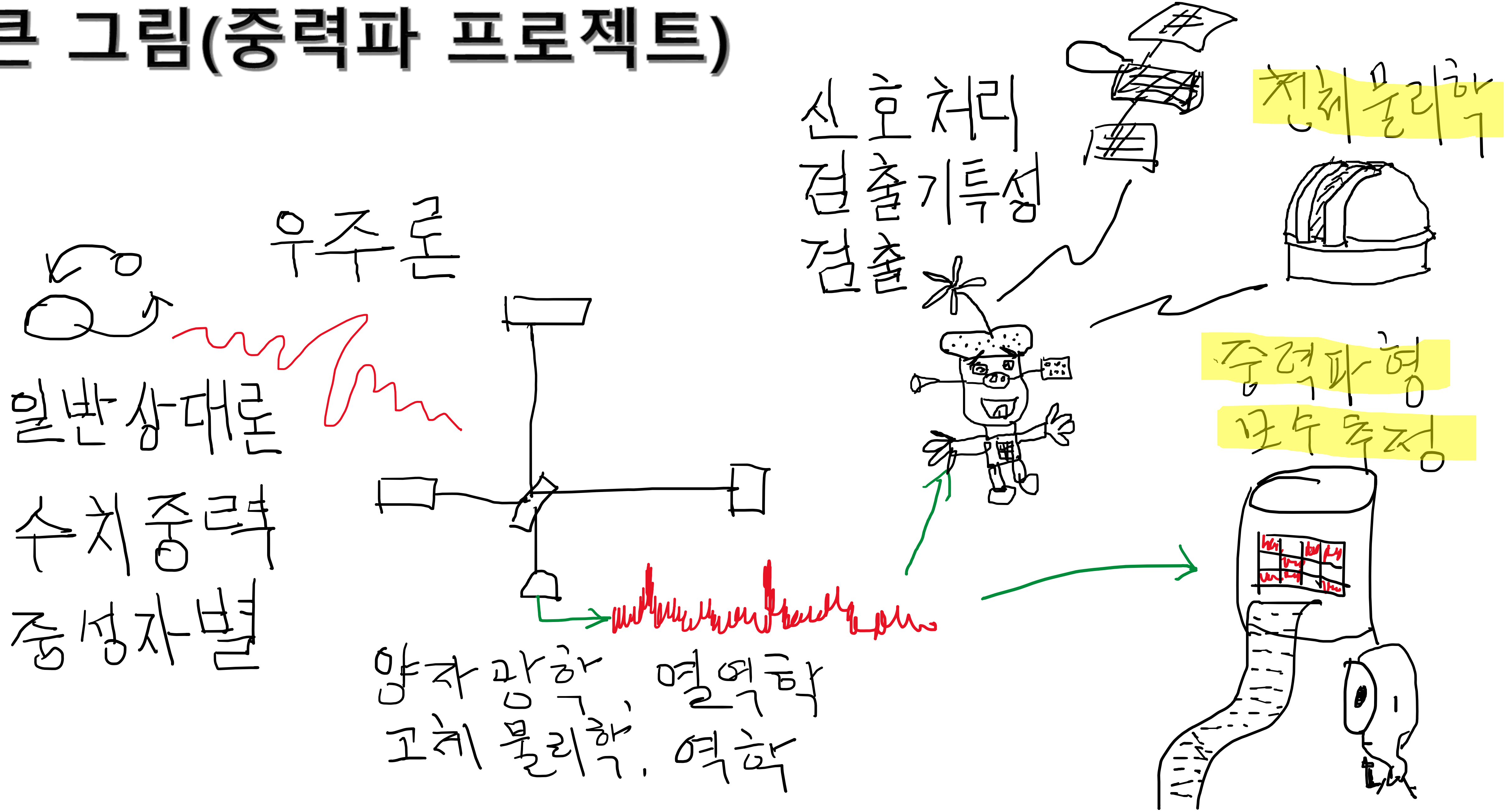
인제대학교 이형원
인제대학교 김정초, 이화여자대학교 김정리 공동연구
2022 수치상대론 및 중력파 겨울학교 / 계산천체물리 경진대회
2022년 1월 18일

목차

- 큰 그림
- 지금까지의 성과
- 중력파 데이터 분석
- 분석 도구
- 후처리 도구
- 맺음 말

큰 그림

큰 그림(중력파 프로젝트)



우주론
일반상대론
수치중력
중성자별
신호처리
검출기특성
검출
천체물리학
중력파형
모수추정
양자광학, 열역학
고체물리학, 역학

중력파 발생

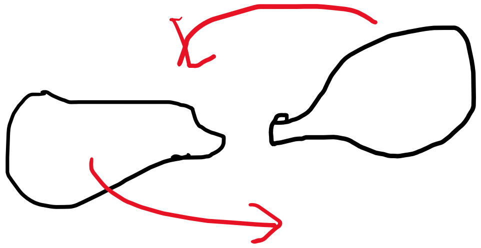
- 아인스타인 방정식
- 선형근사
- 포스트뉴터니안 근사
- 수치중력
- 중성자별 상태 방정식

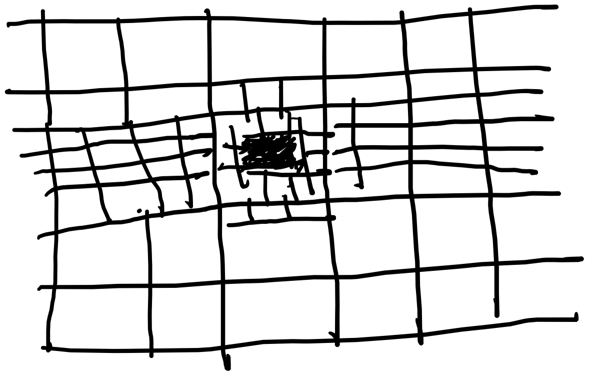
1916

$$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$$

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}$$

$$\frac{GM}{Rc^2} \sim \frac{v^2}{c^2} \ll 1$$





계산천체물리 강의 2
(김진호)

우주론

우주론 수치모의 실험 (김용휘)

- 공변거리(comoving distance)
- 고유거리(proper distance)
- 광도거리(luminosity distance)

FRW 계량

$$c^2 d\tau^2 = c^2 dt^2 - a^2(t)\left(\frac{dr^2}{1-kr^2} + r^2 d\Omega^2\right)$$

$$\chi = \int_{t_{src}}^{t_{obs}} \frac{c\,dt'}{a(t)} = c\int_0^z \frac{dz'}{H(z')}$$

$$D = a(t_{obs})\chi$$

$$d_L = (1+z)D$$

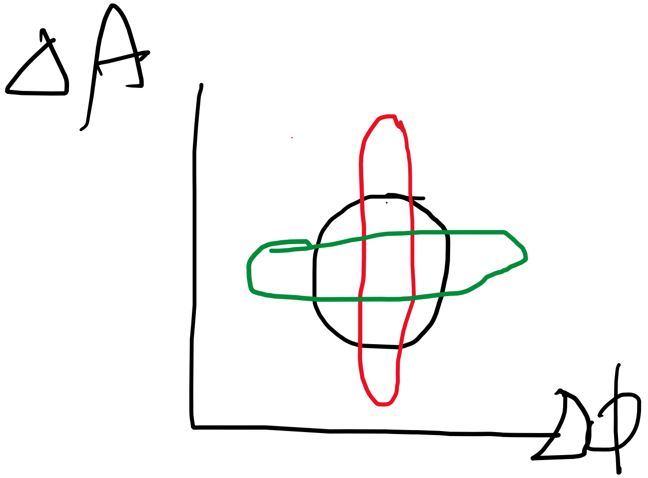
$$H^2 = \frac{8\pi G}{3}\rho$$

$$\tilde{h}(f_{obs}) = (1+z)\tilde{h}(f_{src})$$

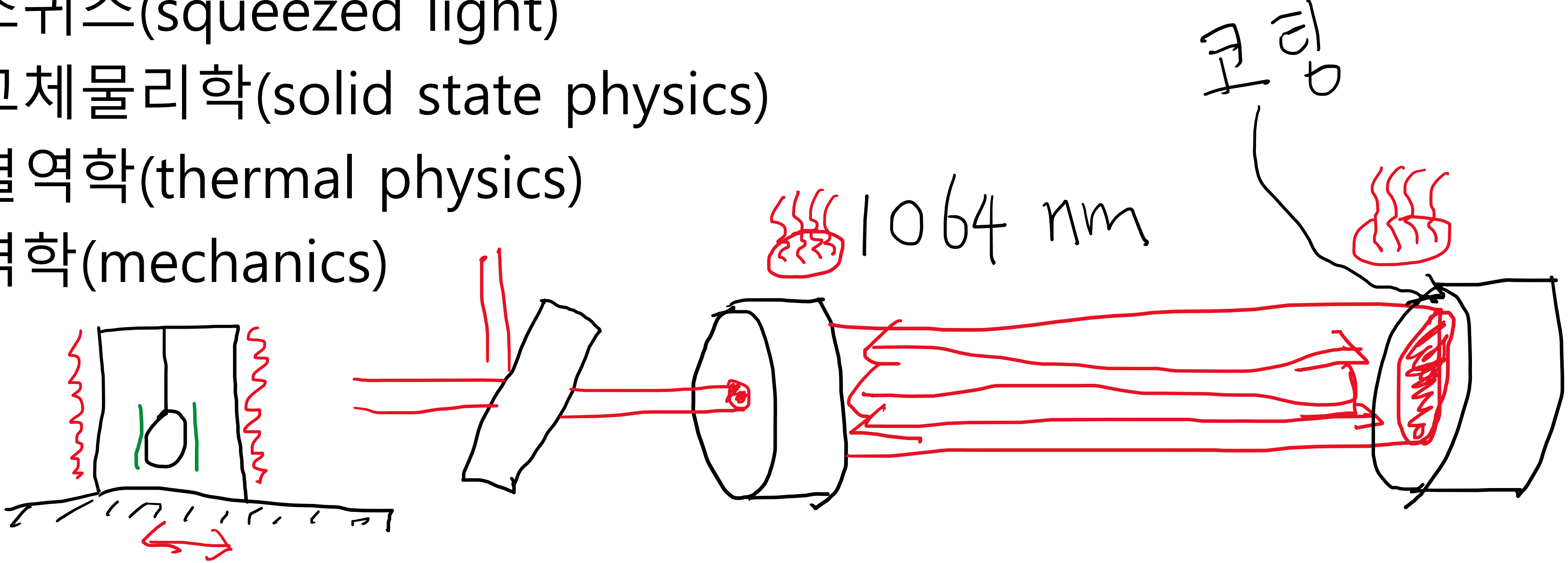
$$\approx \frac{1}{d_L}\underbrace{((1+z)M)}_{M_{det}}{}^{5/6} f^{-\frac{7}{6}} e^{i\Psi(f)}$$

$f \to f_{obs}$

검출기



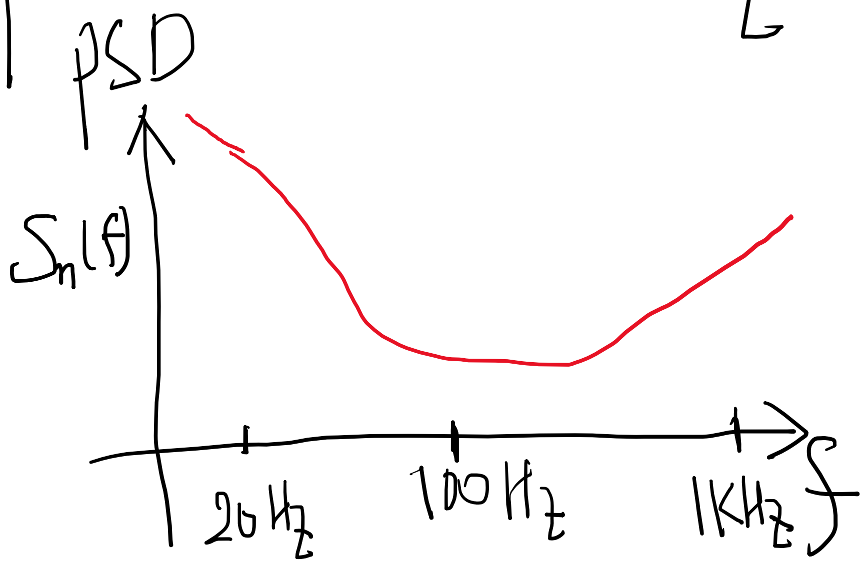
- 양자광학(quantum optics)
- 광공진기(optical cavity)
- 스퀴즈(squeezed light)
- 고체물리학(solid state physics)
- 열역학(thermal physics)
- 역학(mechanics)

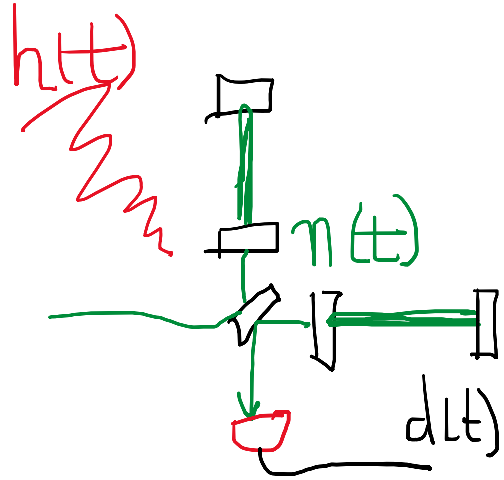


신호처리

기계학습을 이용한 중력파 데이터 분석
(김경민)

- 정합필터(Matched Filter)
- 전력스펙트럼밀도(Power Spectral Density)
- 검출(Search)
- 신호대잡음비(Signal to Noise Ratio)





$$\Lambda(H_1|S) = e^{(s,h)-\frac{1}{2}(h,h)}$$

$$(s,h) = 4\,\mathrm{Re}\int_0^\infty \frac{\tilde{s}(f)\,\tilde{h}^*(f)}{S_n(f)}\,df$$

$$\rho^2 = (h,h), \quad \rho = \frac{x}{\sigma} \quad \text{template}$$

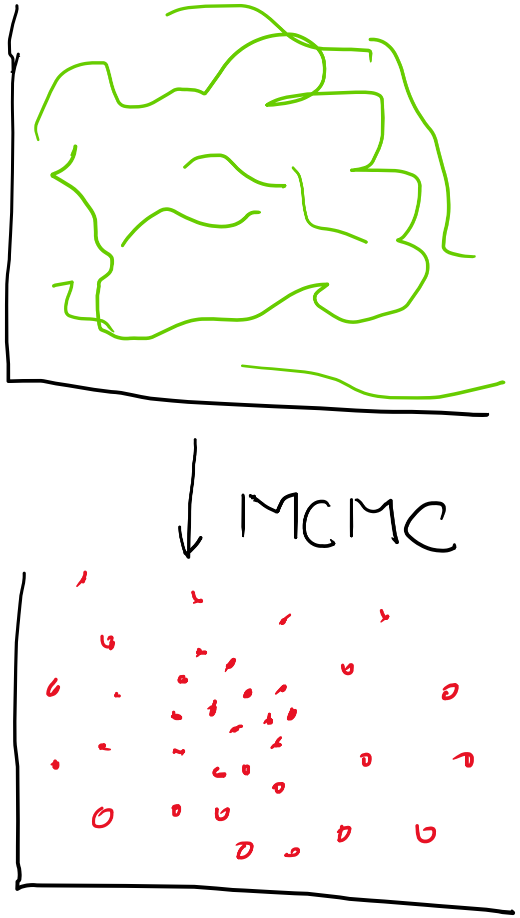
$$d(t) = n(t) + h(t) \qquad \sigma^2 = (g,g)$$

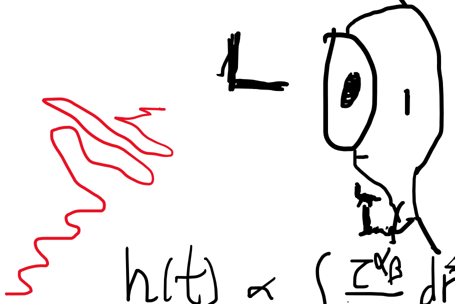
$$h(t) = A g(t), \quad x = (s,g)$$

모수추정(Parameter Estimation)

중력파 데이터 분석 (이형원)	계산천체물리 강의 1 (김영민)

- 중력파형(Waveforms)
- 통계처리(Statistical Analysis)
- 베이지언 추정(Bayesian Inference)
- 몬테카를로(Markov Chain Monte Carlo)





$$h(t) \propto \int \frac{\ddot{I}^{ab}}{r} d\vec{r} \sim \frac{\ddot{I}}{r}$$

↓ FFT

$$\tilde{h}(f) \sim \frac{M_c^{5/6}}{r} f^{-\frac{7}{6}} e^{i\psi(f)}$$

$$d(t) = n(t) + h(t)$$

$$\mathcal{L} \sim e^{-\frac{1}{2}(d-h,\, d-h)}$$

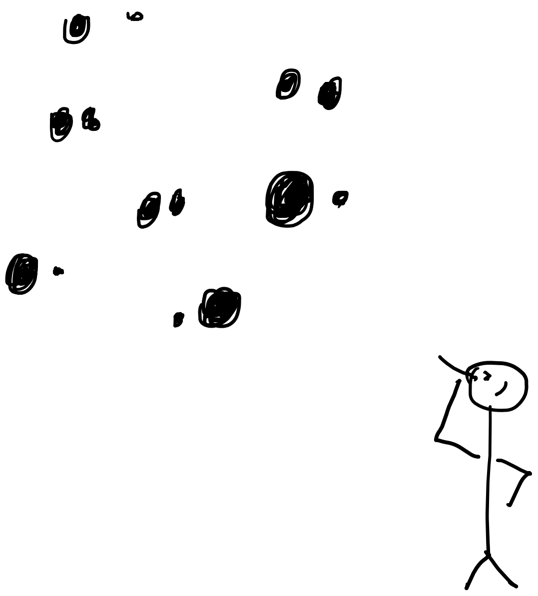
$$p(\vec{\theta}\,|\,\vec{d}) \sim \mathcal{L}(\vec{d}\,|\,\vec{\theta})\, p(\vec{\theta})$$

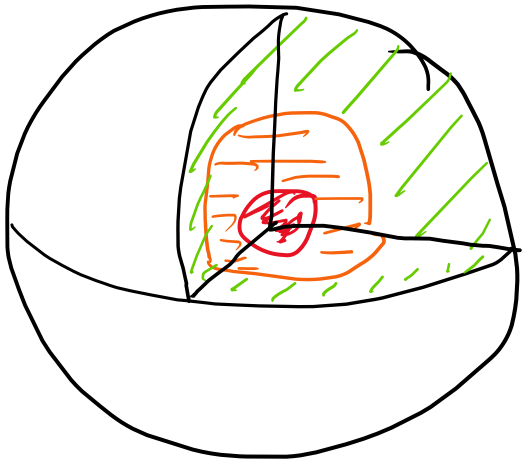
천체물리학(Astrophysics)

중성자별의 내부 구조 및 상태 방정식
(이창환)

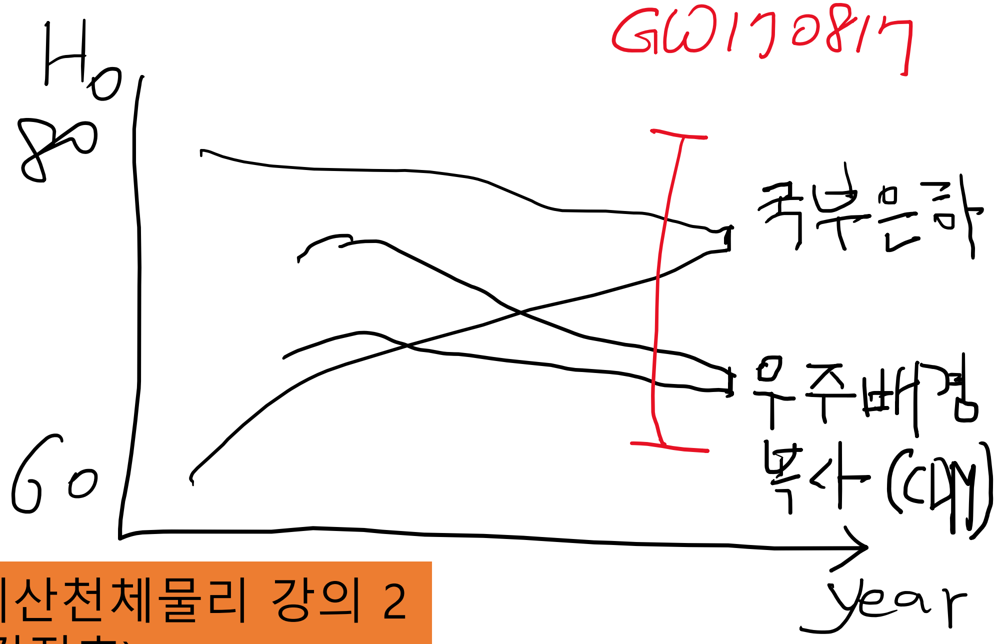
성단 역학 및 블랙홀 쌍성의 형성
(배영복)

- 블랙홀 병합율(Blackhole Merger rate)
- 중성자별 상태 방정식(Equation of State of NS)
- 허블 상수(Hubble Constant)





TOV 방정식

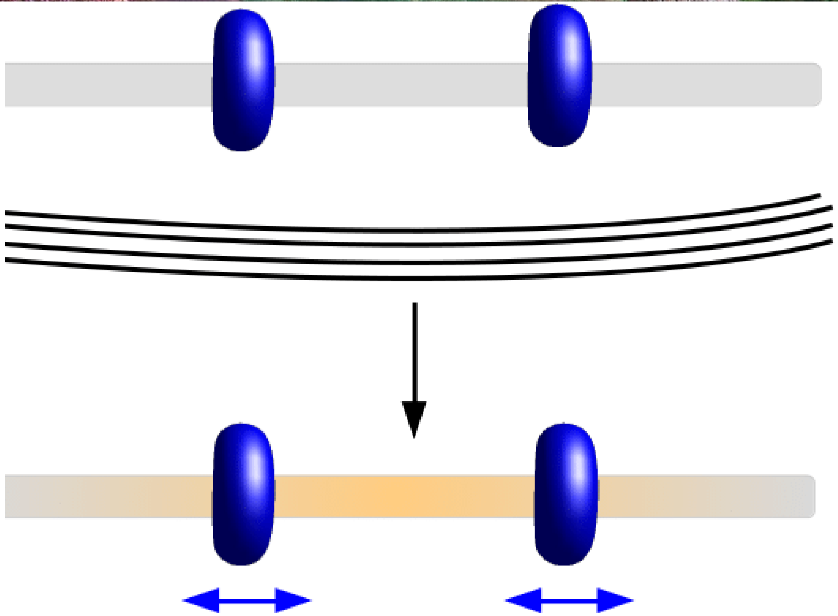


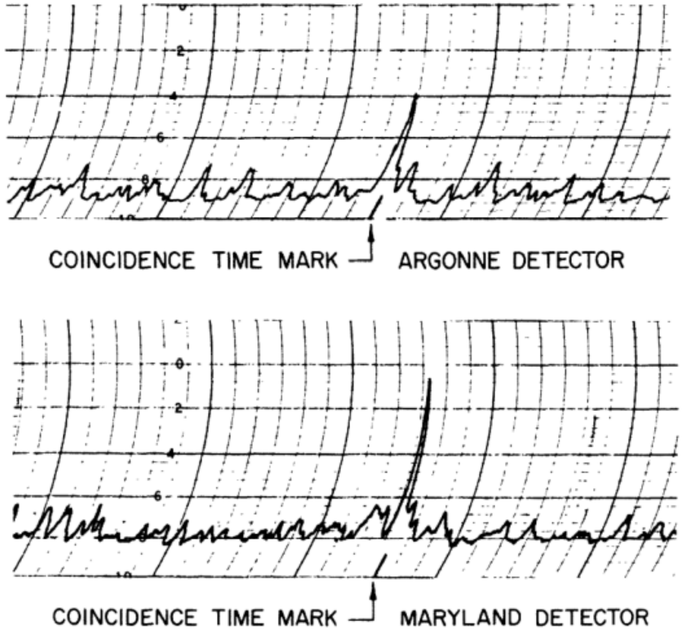
계산천체물리 강의 2
(김진호)

지금까지의 성과

초기 검출기(Weber Bar)

- Chapel Hill Conference, 1957
- Feynmann's Sticky bids
- Weber bar(1966)
- Braginsky, Thorne(1972)



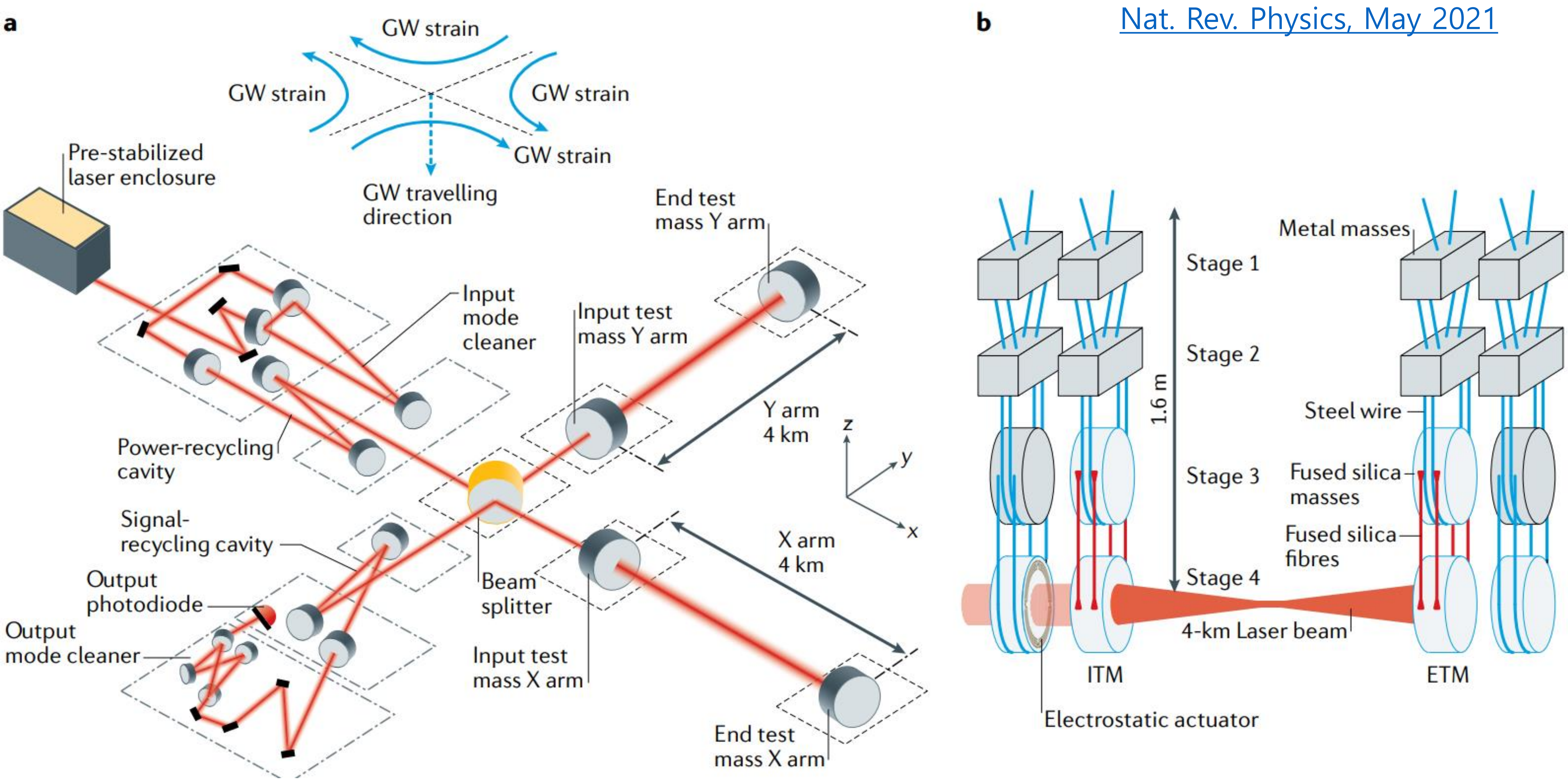


간섭계(Laser Interferometer)

- Gertsenshtein and Pustovoit, 1962
- Weber and Weiss, 1967
- Weiss, noise analysis, 1972
- Drever and Whitcomb, Caltech, 1980
- Drever, Weiss and Thorne, LIGO, 1984
- Barish, LIGO Director, 1997
- Virgo joined, 2007
- KAGRA joined, 2019

LIGO 검출기

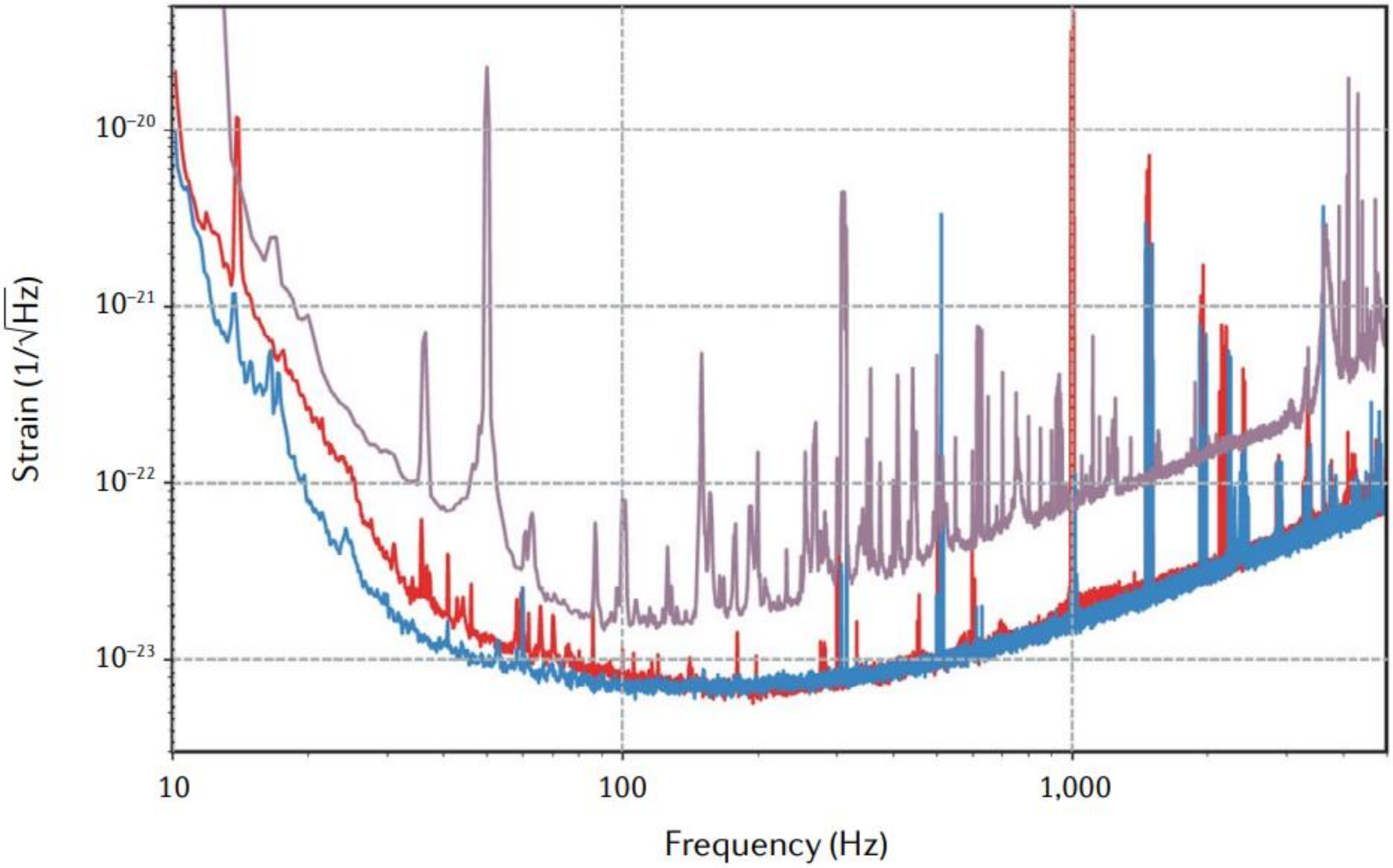
Nat. Rev. Physics, May 2021



[Nat. Rev. Physics, May 2021]

LIGO 감도

- 지진잡음
- 열 잡음
- 중력경사 잡음
- 양자 잡음
- 기술 잡음
  - 레이저 주파수
  - 레이저 세기
  - 레이저 산란
  - 감지기, 작동기
  - 전자기 잡음
  - 입자(우주선)



공공 알림(Public Alerts)


https://gcn.gsfc.nasa.gov/gcn3_archive.html

GCN Circulars Archive (in serial number order)

This page changes after each Circular submission, so hit the <reload> button NOW.

The processing of Circulars to this page was stopped (from ~01:30 to ~18:00 UT 01 Dec 2020).
The Circulars page has been fixed as of ~18:00 UT 01 Dec 2020.
(The Circulars WERE ALWAYS BEING DISTRIBUTED -- it is only this archive web page (and sub-pages) that had problems.)

1. The Latest Circulars
2. Older Circulars
3. Tarfile of all Circulars
4. Circulars grouped by each Event
5. All Circulars on the GRB source type
6. All Circulars on the GW source type
7. All Circulars on the SGR source type
8. All Circulars on the misc source type (ie not in any of the above 3 pages).

https://emfollow.docs.ligo.org/userguide/

LIGO Scientific Collaboration

VIRGO

Public Alerts User Guide

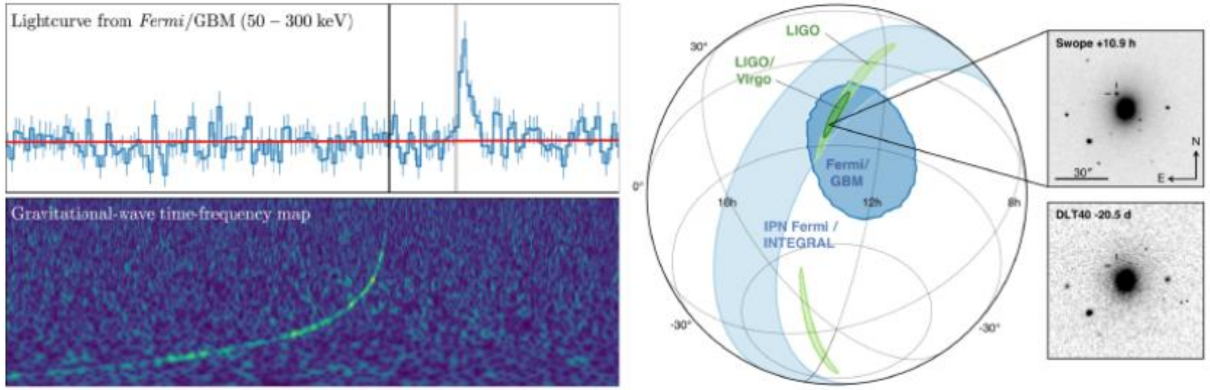
Primer on public alerts for astronomers from the LIGO and Virgo gravitational-wave observatories.

Navigation

- Getting Started Checklist
- Observing Capabilities
- Data Analysis
- Alert Contents
- Sample Code
- Additional Resources
- Early-Warning Alerts

Getting Started Checklist →

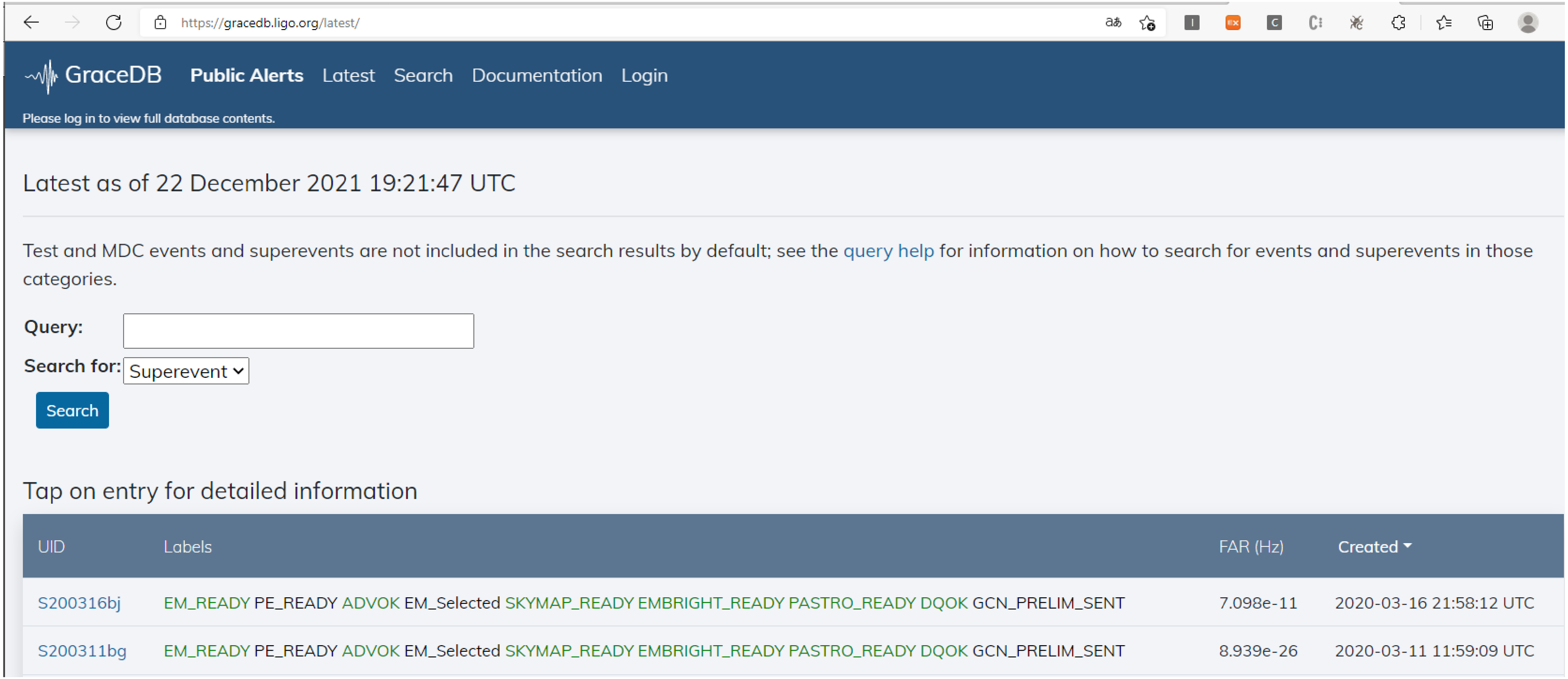
LIGO/Virgo Public Alerts User Guide ¶



Welcome to the LIGO/Virgo Public Alerts User Guide! This document is intended for both professional astronomers and science enthusiasts who are interested in receiving alerts and real-time data products related to gravitational-wave (GW) events.

Three sites (LHO, LLO, Virgo) together form a global network of ground-based GW

GraceDB


https://gracedb.ligo.org/latest/
GraceDB
Public Alerts
Latest
Search
Documentation
Login
Please log in to view full database contents.
Latest as of 22 December 2021 19:21:47 UTC
Test and MDC events and superevents are not included in the search results by default; see the query help for information on how to search for events and superevents in those categories.
Query:
Search for: Superevent
Search
Tap on entry for detailed information
UID
Labels
FAR (Hz)
Created
S200316bj
EM_READY PE_READY ADVOK EM_Selected SKYMAP_READY EMBRIGHT_READY PASTRO_READY DQOK GCN_PRELIM_SENT
7.098e-11
2020-03-16 21:58:12 UTC
S200311bg
EM_READY PE_READY ADVOK EM_Selected SKYMAP_READY EMBRIGHT_READY PASTRO_READY DQOK GCN_PRELIM_SENT
8.939e-26
2020-03-11 11:59:09 UTC

검색엔진

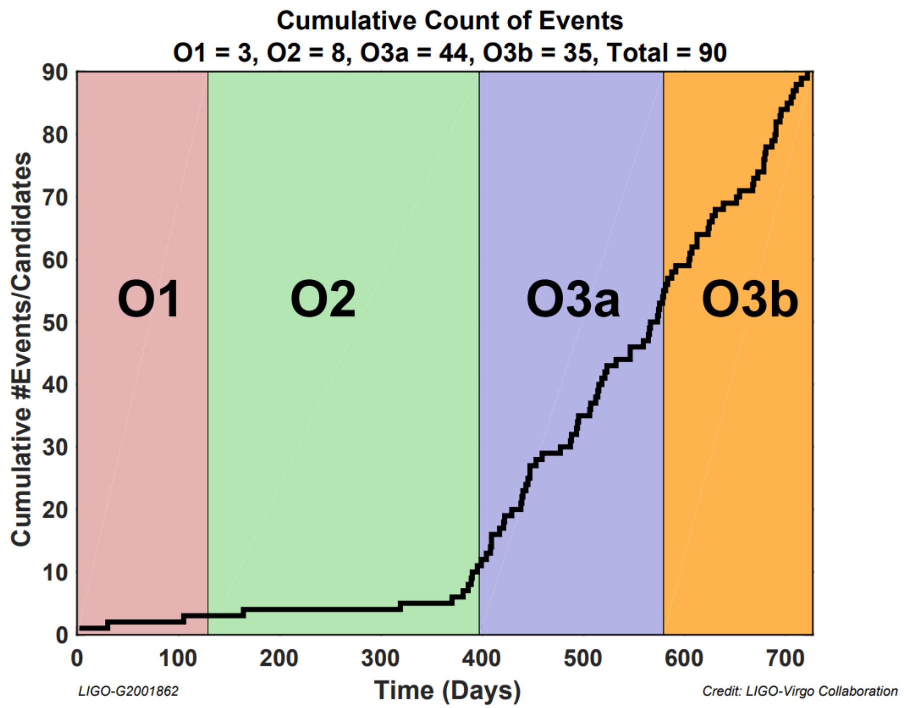
- 파형의존
  - GstLAL: Gstreamer from LALSuite
  - MBTA: Multi-Band Template Analysis
  - PyCBC Broad
  - PyCBC BBH
- 파형독립
  - cWB: Coherent Wave-Burst

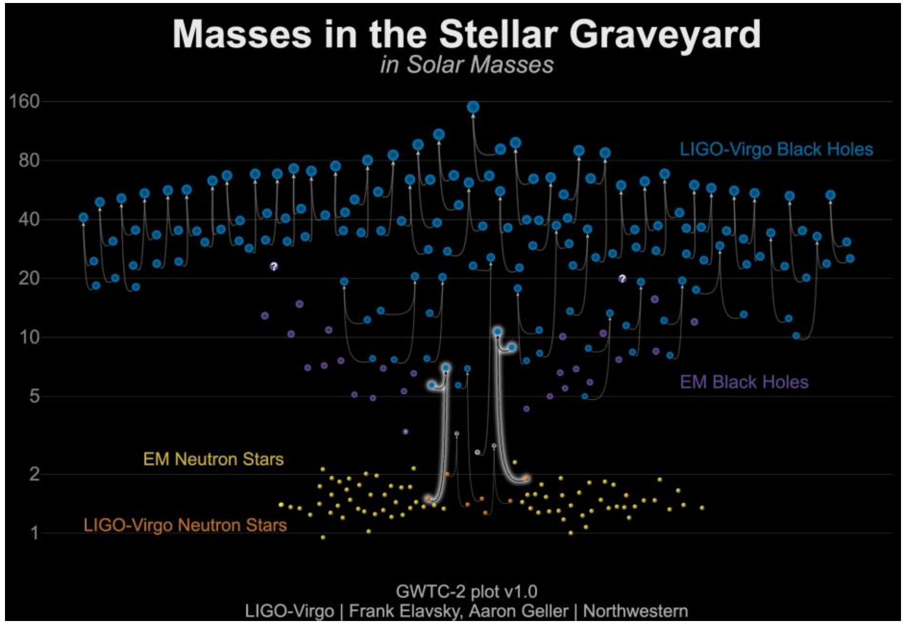
알려진 CBC중력파형 사용
정합필터와 파형집합 사용
검출기 사이의 동시성 확인
GstLAL은 한 검출기 검색 가능

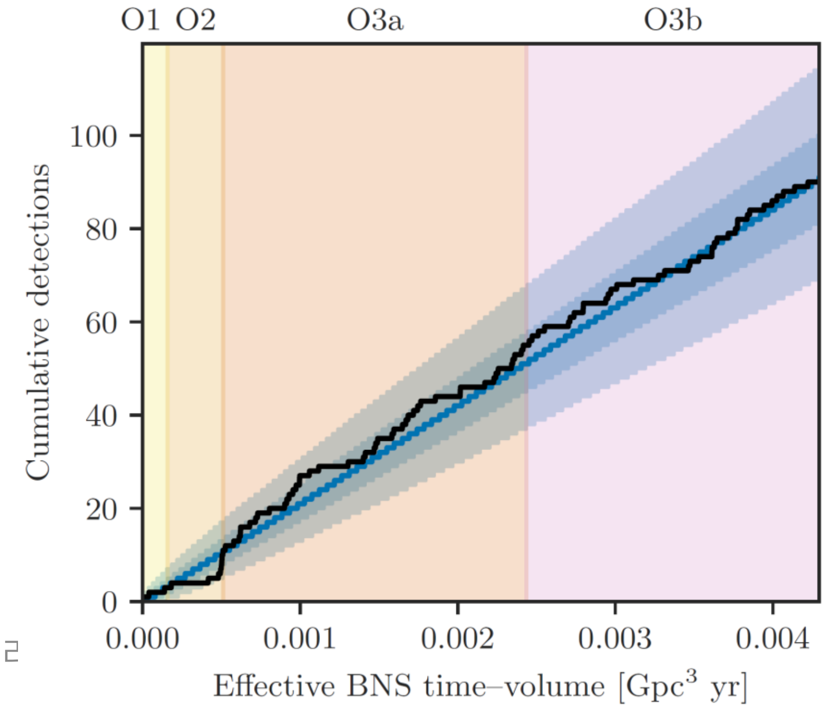
CBC이외의 중력파 검출 가능
정합필터 사용하지 않음
검출기 사이의 동시성 확인
초과 에너지 계산

관측결과(Observations)

- O1: 2015/09/12~2016/01/19
- O2: 2016/11/30~2017/08/25
- O3a : 2019/04/01~2019/09/30
- O3b : 2019/11/01~2020/03/27







관측결과(90)

GWTC-3

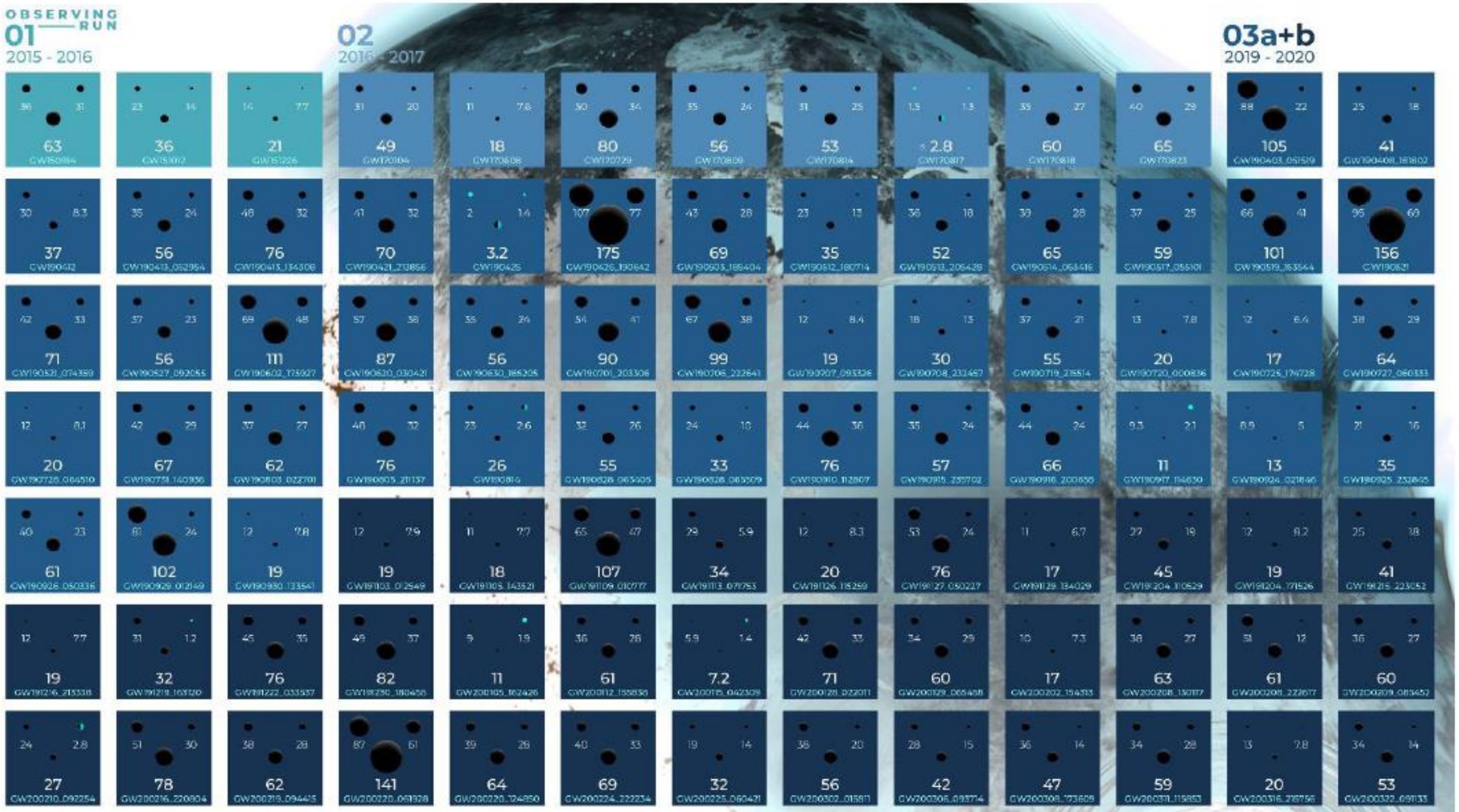
The gravitational-wave story

1916 Einstein predicts gravitational waves in general relativity

1974 First indirect evidence of gravitational waves from binary pulsars

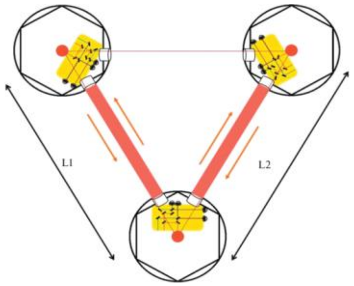
2015 First observation of gravitational waves at the start of O1

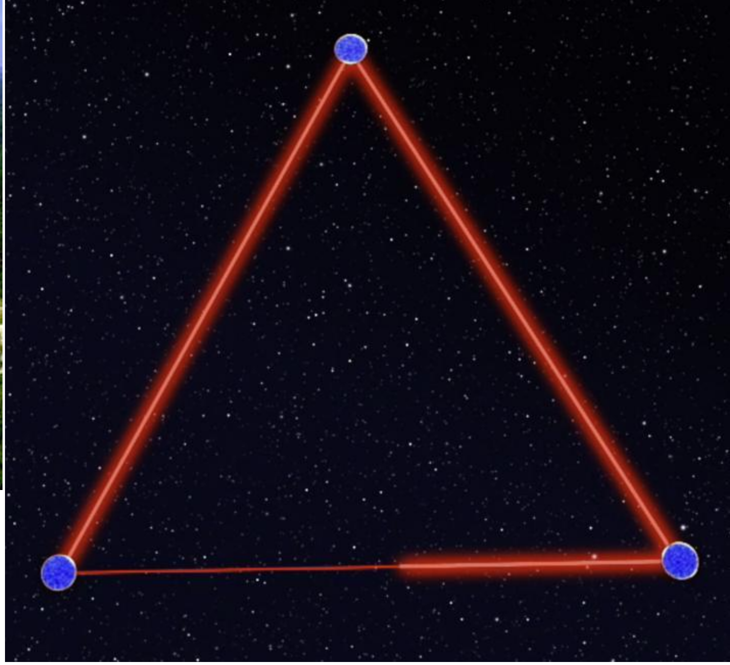
Observing runs
O1: 2015–2016
O2: 2016–2017
O3: 2019–2020
O4: ~2022–2023

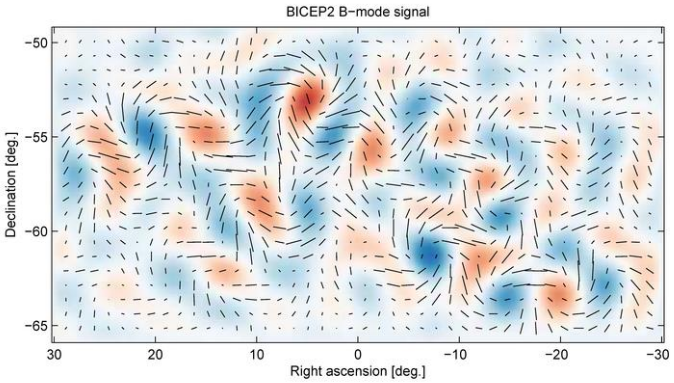


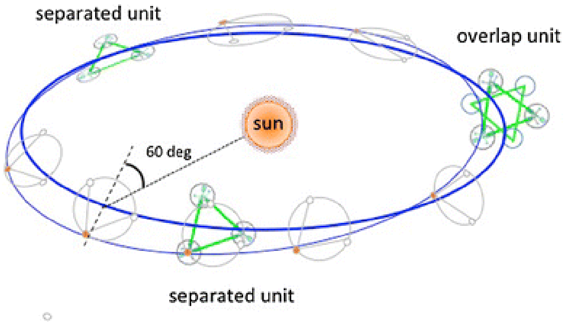
여러가지 검출기

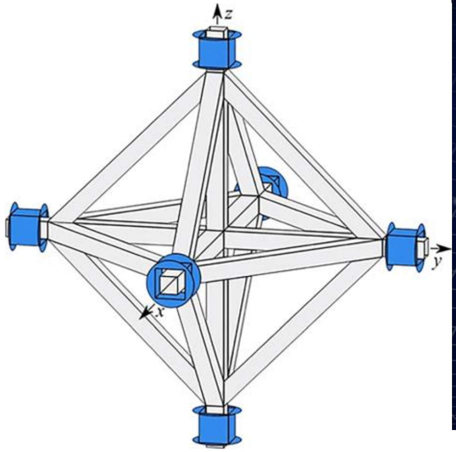
- 공명기
- 지상간섭계
- 우주간섭계
- 펄사타이밍어레이
- 우주배경복사 편광





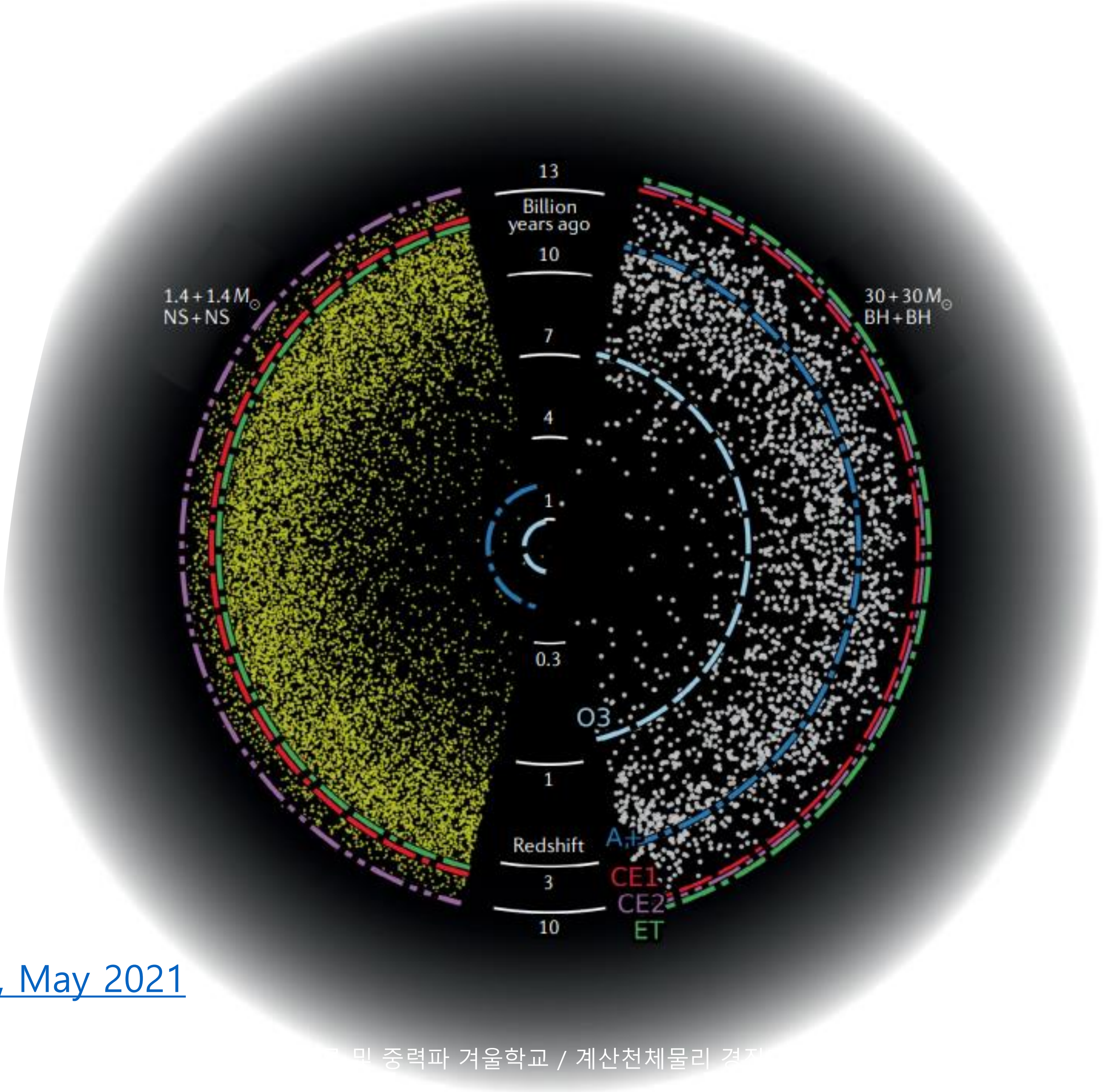






검출기 미래

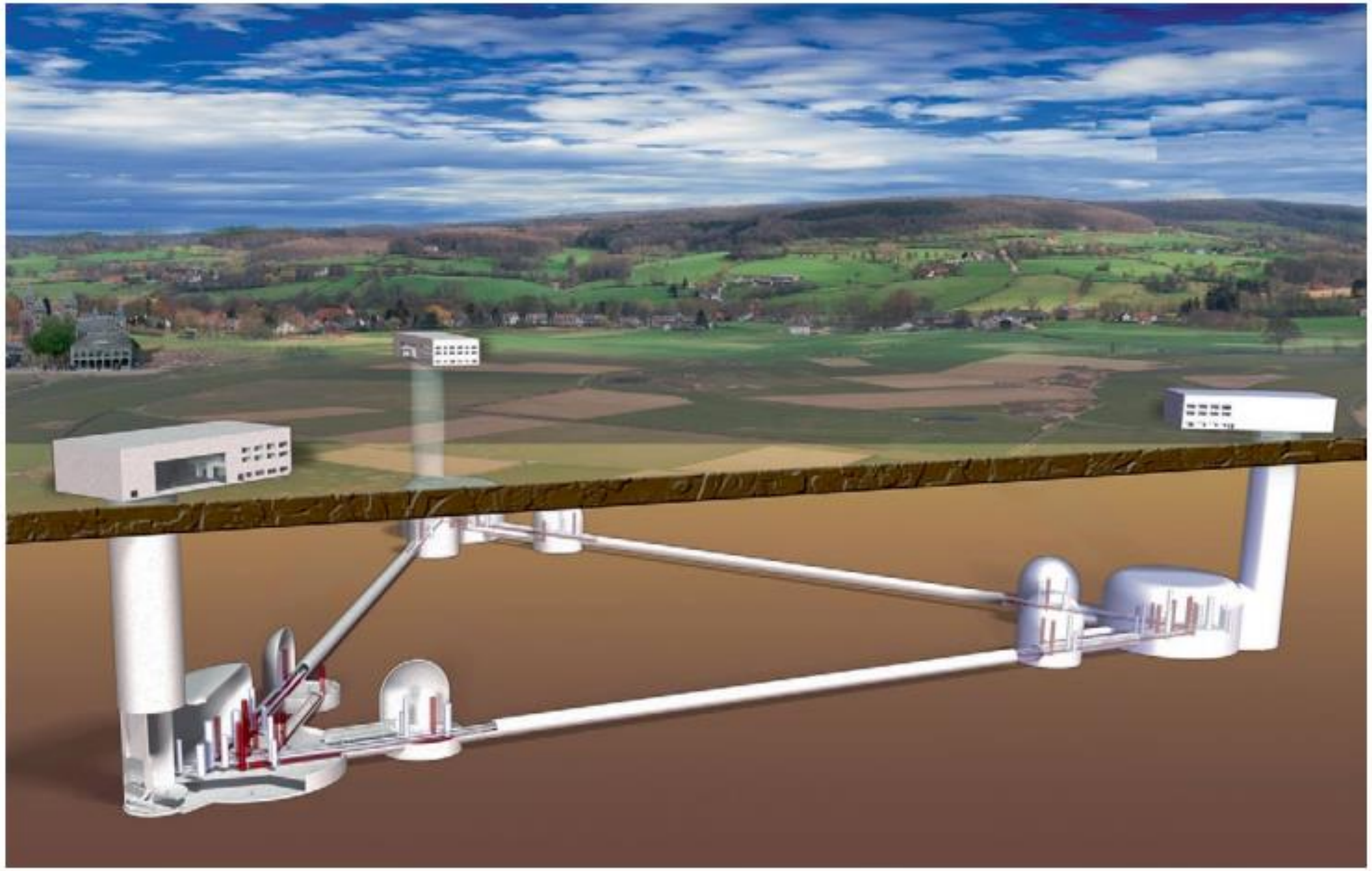
- O3/O4/O5
- A+(Advanced LIGO)
- CE1, CE2(Cosmic Explorer)
- ET(Einstein Telescope)



[Nat. Rev. Physics, May 2021]

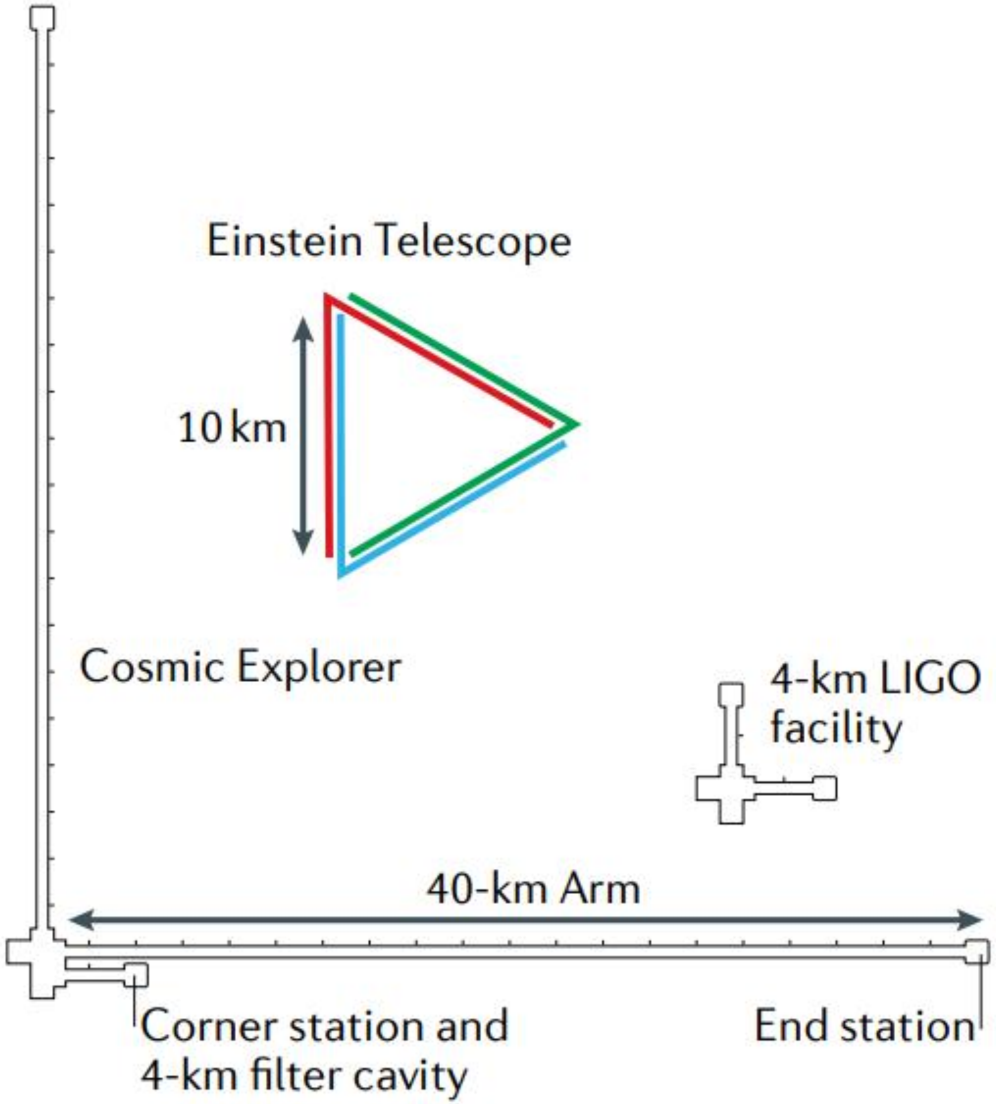
3세대 검출기

a



[Nat. Rev. Physics, May 2021]

b



중력파 천체물리학

Box 1 | **Fundamental questions addressed through gravitational-wave observations**

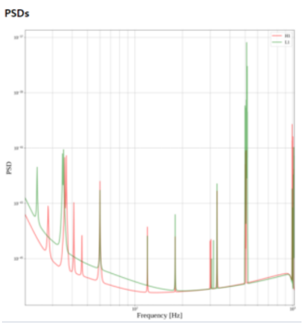
- What is the physics of stellar core collapse? How often do core-collapse supernovae occur[257,258]?
- What is the equation of state, and what are the radii, of neutron stars[120,121]?
- What are the multi-messenger emission mechanisms of high-energy transients (gamma-ray bursts and kilonovae)[259]?
- How do binary black holes of tens of solar masses form and evolve[260,261]?
- How did super-massive black holes at the cores of galactic nuclei form and evolve, and what were their seeds and demographics[262]?
- Are black hole spacetimes as predicted by general relativity[136]?
- Are there any signatures of horizon structure or other manifestations of quantum gravity accessible to gravitational-wave observations[263]?
- Is dark matter composed, in part, of primordial black holes, or must it be composed solely from exotic matter such as axions or dark fermions[263]?
- What is the expansion rate of the Universe[264]?
- What is the nature of dark energy[264]?
- Is there a measurable gravitational-wave stochastic background due to phase transitions in the early Universe? If so, what were its properties[265–267]?
- How does gravity behave in the strong/highly dynamical regime[99–102]?
- Do we live in a Universe with large extra dimensions[116]?
- Are black holes, neutron stars and white dwarfs the only compact objects in our Universe, or are there even more exotic objects[133]?

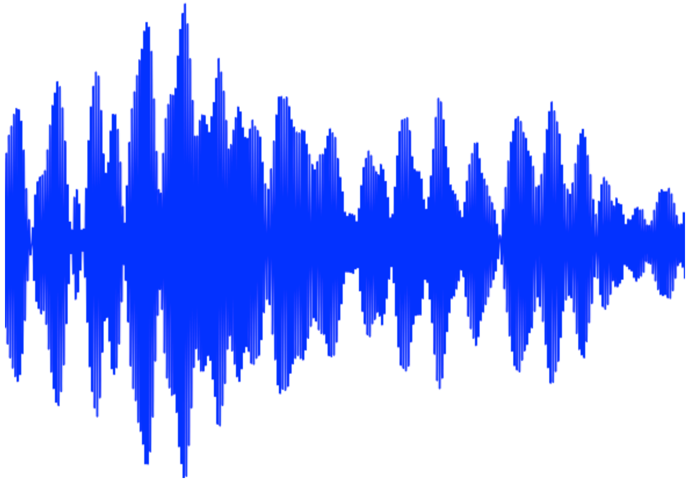
Nat. Rev. Physics, May 2021

중력파 데이터 분석

데이터분석

- 고유특성
  - $m_1, m_2, \vec{S}_1, \vec{S}_2, \Lambda_1, \Lambda_2, e_0$
- 외부특성
  - $t_c, \varphi_c, \alpha, \delta, D, \theta_{JN}, \psi$

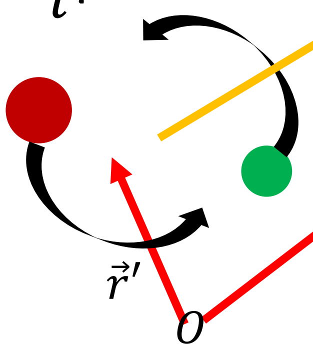


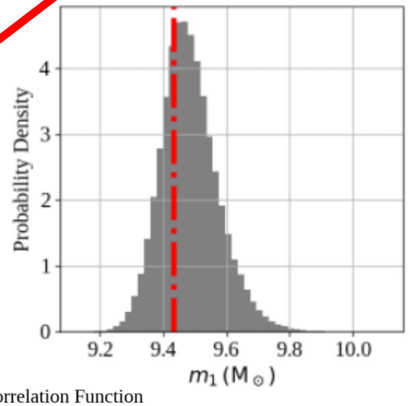


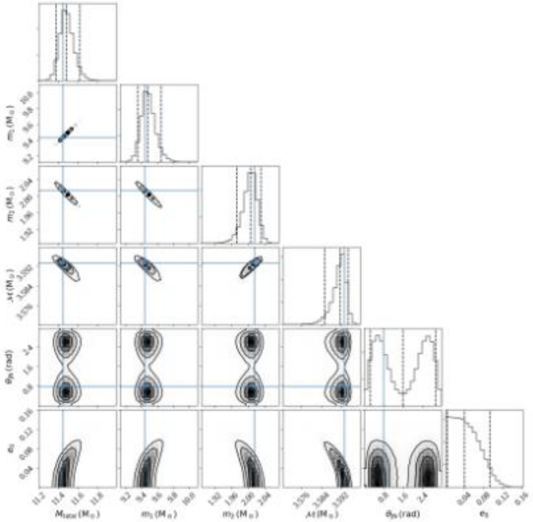
$\bar{h}^{\mu\nu}$

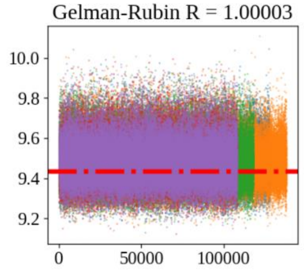
$\vec{R} = \vec{r} - \vec{r}'$

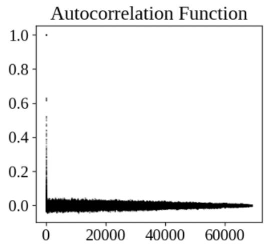
$\vec{r}$











얼마나 비슷한가?
Likelihood

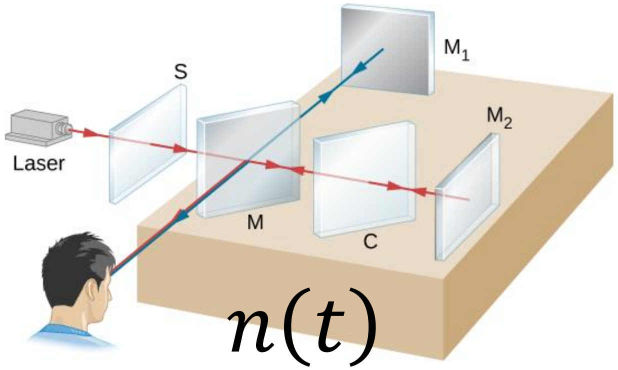
베이지언 추론
Posterior

샘플생성
MCMC, Nested, RIFT

적분
매개변수 분포

Linear Optimal Filter

$h(t)$



$n(t)$

$d(t) = h(t) + n(t)$

$\varepsilon(t) = \mathcal{H}\big(d(t)\big) - h(t)$

$$\text{minimize } \overline{\varepsilon^2}$$

$$K(f) \propto \frac{h(f)}{S_n(f)}$$

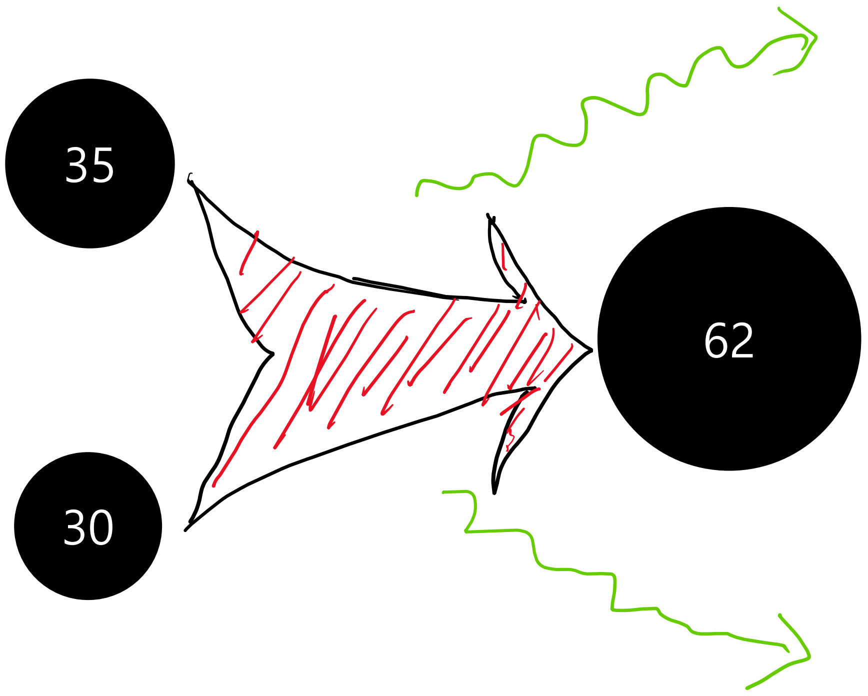
중력파형

- 107 Waveforms
- Post-Newtonian
- Time-domain
- Frequency-domain
- Tidal effects
- Spin effects
- NR approximation
- Effective One Body

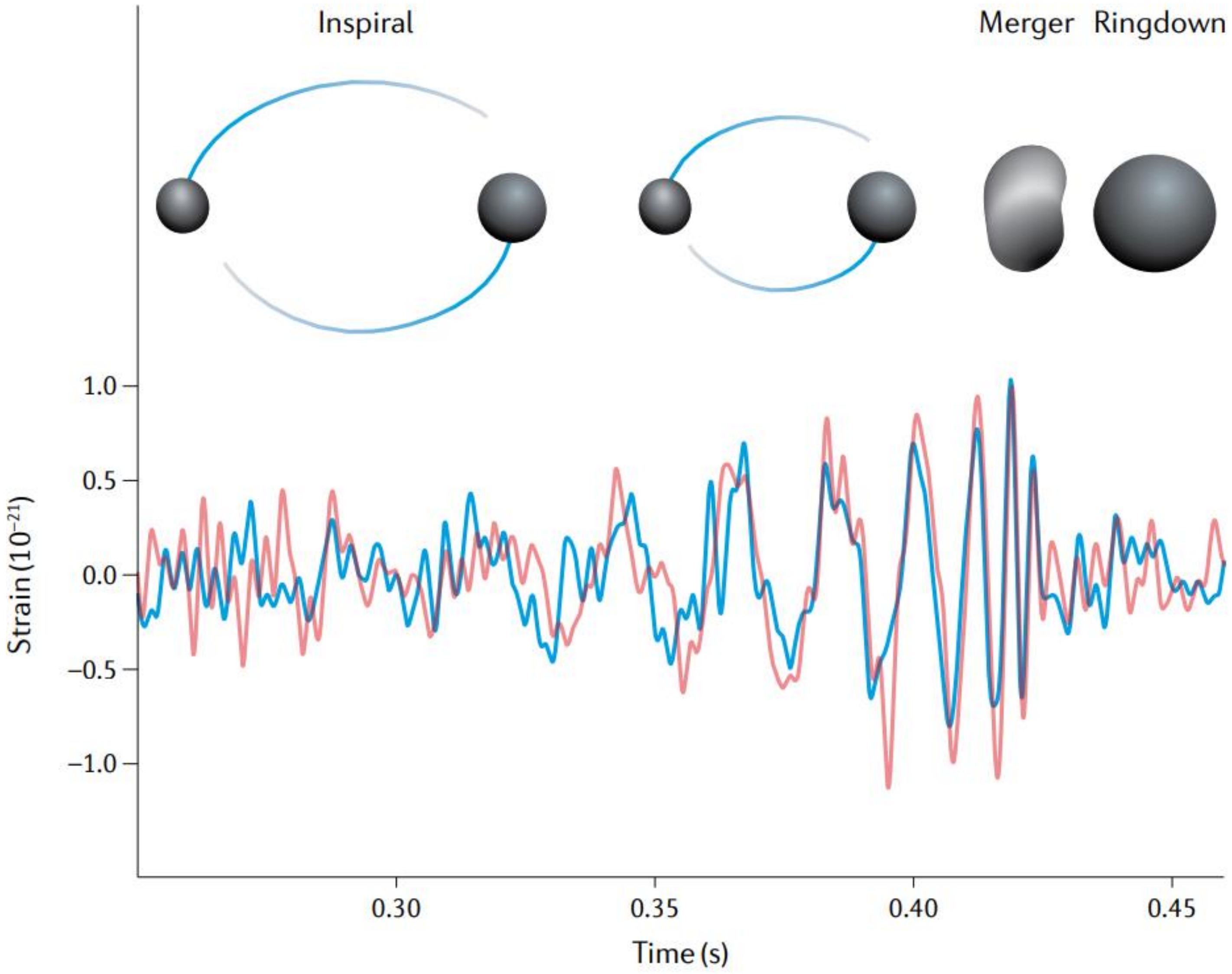
LALSimInspiral.h

```
enum   Approximant {
    TaylorT1, TaylorT2, TaylorT3, TaylorF1,
    EccentricFD, TaylorF2, TaylorF2Ecc, TaylorF2NLTides,
    TaylorR2F4, TaylorF2RedSpin, TaylorF2RedSpinTidal, PadeT1,
    PadeF1, EOB, BCV, BCVSpin,
    SpinTaylorT1, SpinTaylorT2, SpinTaylorT3, SpinTaylorT4,
    SpinTaylorT5, SpinTaylorF2, SpinTaylorFrameless, SpinTaylor,
    PhenSpinTaylor, PhenSpinTaylorRD, SpinQuadTaylor, FindChirpSP,
    FindChirpPTF, GeneratePPN, BCVC, FrameFile,
    AmpCorPPN, NumRel, NumRelNinja2, Eccentricity,
    EOBNR, EOBNRv2, EOBNRv2HM, EOBNRv2_ROM,
    EOBNRv2HM_ROM, TEOBResum_ROM, SEOBNRv1, SEOBNRv2,
    SEOBNRv2_opt, SEOBNRv3, SEOBNRv3_pert, SEOBNRv3_opt,
    SEOBNRv3_opt_rk4, SEOBNRv4, SEOBNRv4_opt, SEOBNRv4P,
    SEOBNRv4PHM, SEOBNRv2T, SEOBNRv4T, SEOBNRv1_ROM_EffectiveSpin,
    SEOBNRv1_ROM_DoubleSpin, SEOBNRv2_ROM_EffectiveSpin, SEOBNRv2_ROM_DoubleSpin, SEOBNRv2_ROM_DoubleSpin_HI,
    Lackey_Tidal_2013_SEOBNRv2_ROM, SEOBNRv4_ROM, SEOBNRv4HM_ROM, SEOBNRv4_ROM_NRTidal,
    SEOBNRv4_ROM_NRTidalv2, SEOBNRv4_ROM_NRTidalv2_NSBH, SEOBNRv4T_surrogate, HGimri,
    IMRPhenomA, IMRPhenomB, IMRPhenomFA, IMRPhenomFB,
    IMRPhenomC, IMRPhenomD, IMRPhenomD_NRTidal, IMRPhenomD_NRTidalv2,
    IMRPhenomNSBH, IMRPhenomHM, IMRPhenomP, IMRPhenomPv2,
    IMRPhenomPv2_NRTidal, IMRPhenomPv2_NRTidalv2, IMRPhenomFC, TaylorEt,
    TaylorT4, EccentricTD, TaylorN, SpinTaylorT4Fourier,
    SpinTaylorT5Fourier, SpinDominatedWf, NR_hdf5, NRSur4d2s,
    NRSur7dq2, NRSur7dq4, SEOBNRv4HM, NRHybSur3dq8,
    IMRPhenomXAS, IMRPhenomXHM, IMRPhenomPv3, IMRPhenomPv3HM,
    IMRPhenomXP, IMRPhenomXPHM, TEOBResumS, IMRPhenomT,
    IMRPhenomTHM, IMRPhenomTP, IMRPhenomTPHM, NumApproximants
}
```

GW150914



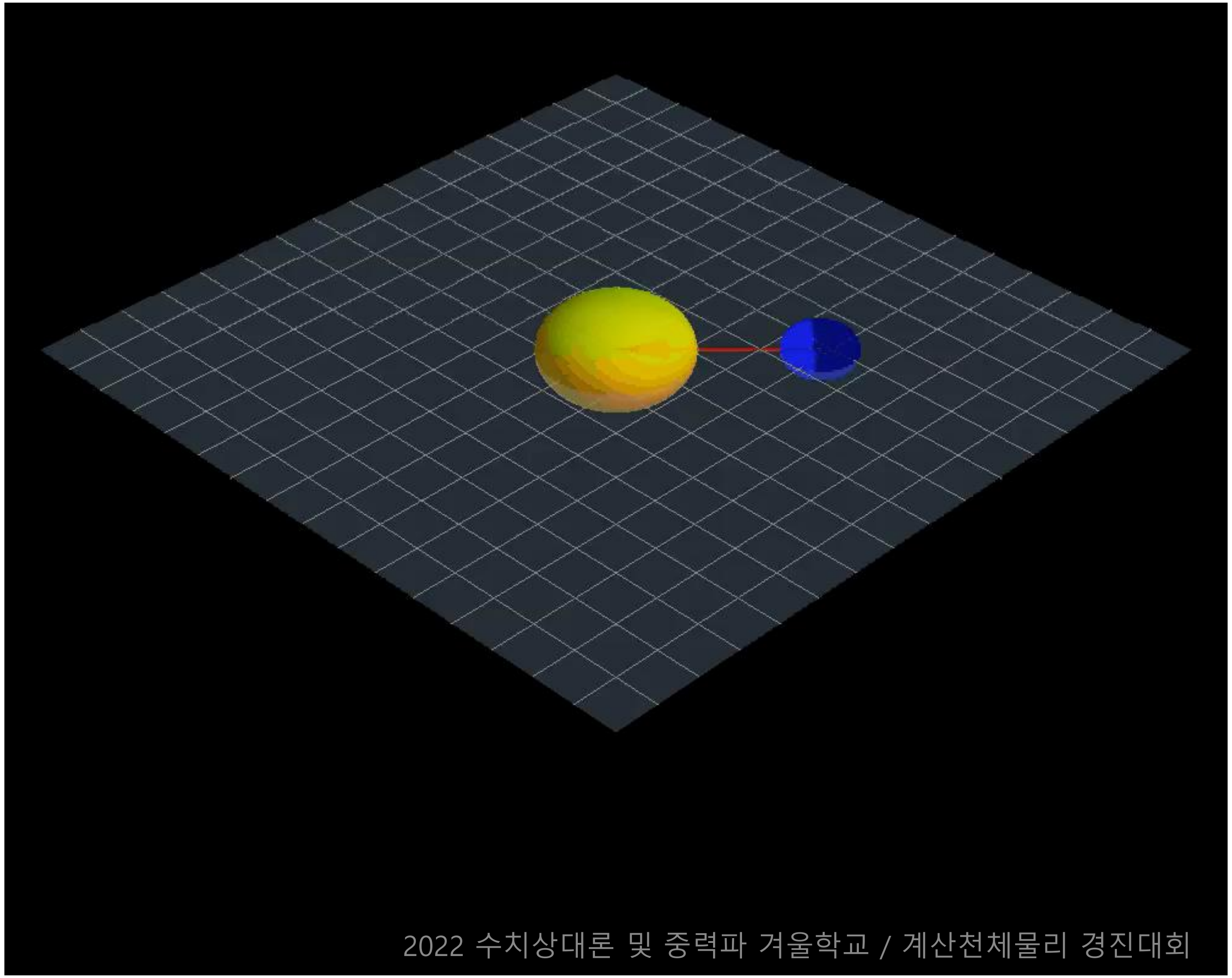
35
30
62



Inspiral
Merger
Ringdown
1.0
0.5
0.0
−0.5
−1.0
Strain (10^{-21})
0.30
0.35
0.40
0.45
Time (s)

Binary orbit in Newtonian Theory

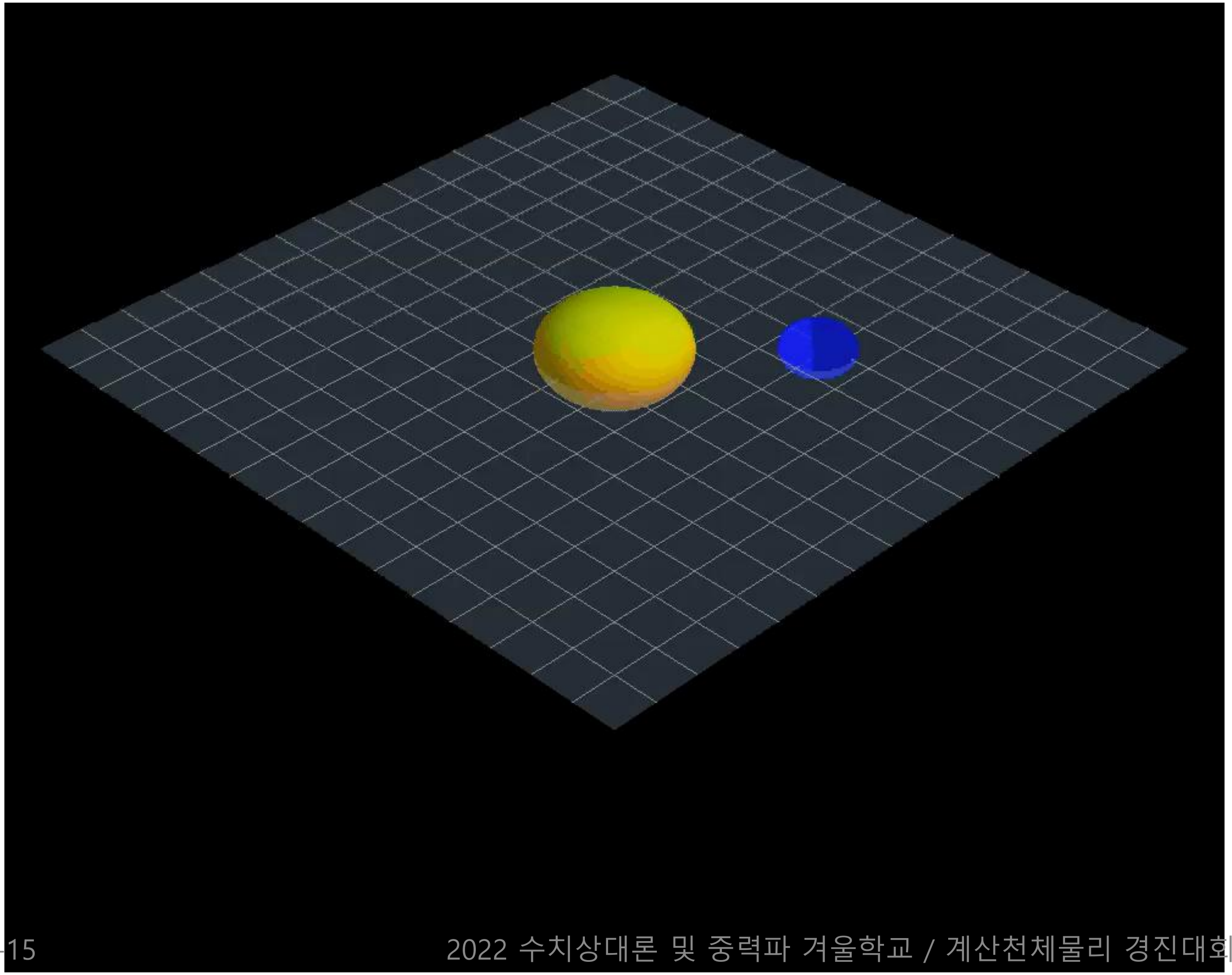
Orbital motion does not change space-time



Credit: Carl Rodriguez

General Relativity

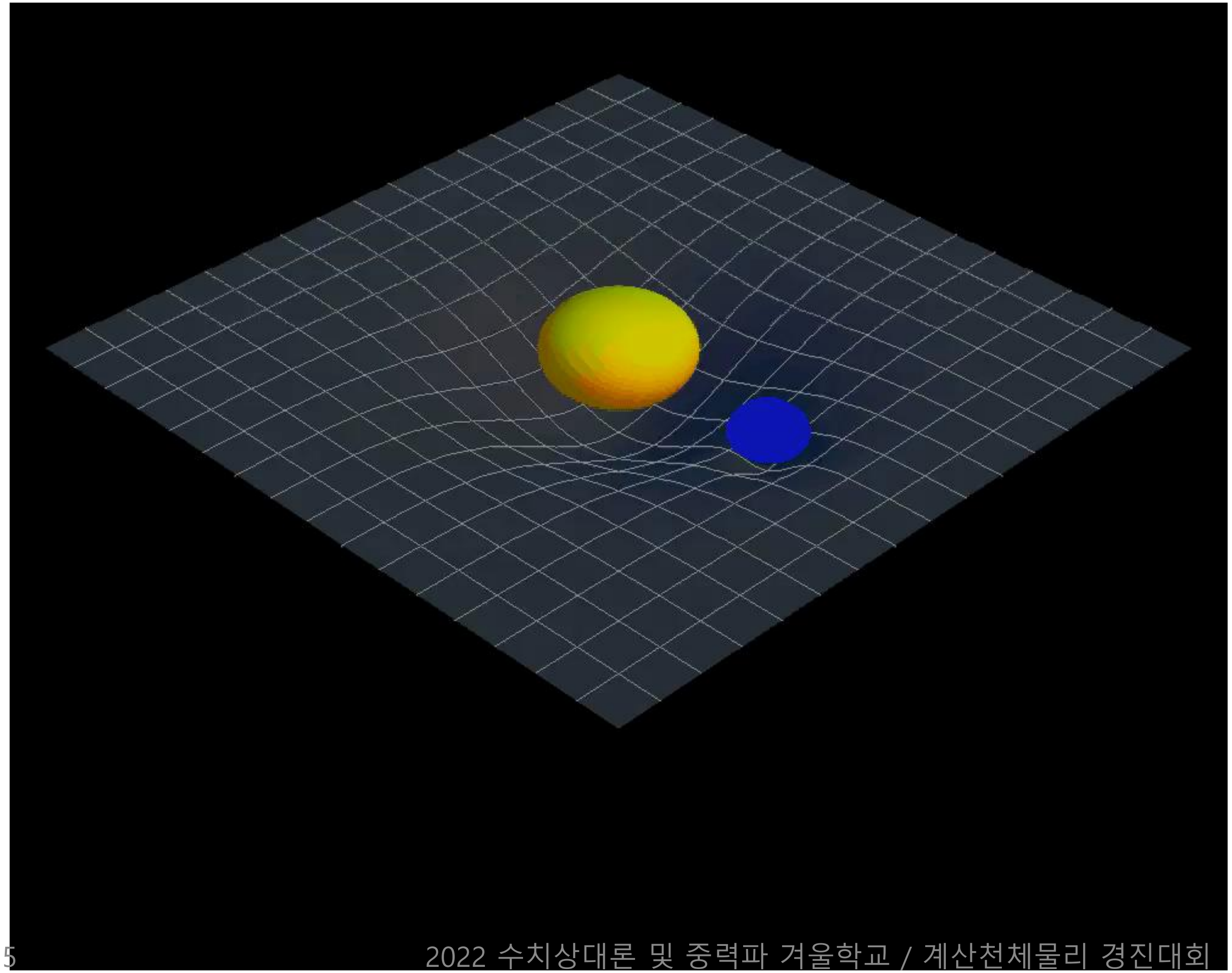
Mass, Energy ~ curved space-time



Credit: Carl Rodriguez

아인슈타인의 상대성 이론과 쌍성의 궤도 운동

Change in space-time caused by binary motion propagate in speed of light(GW)



Credit: Carl Rodriguez

중력파 자료 예

KAGALI h(t) data

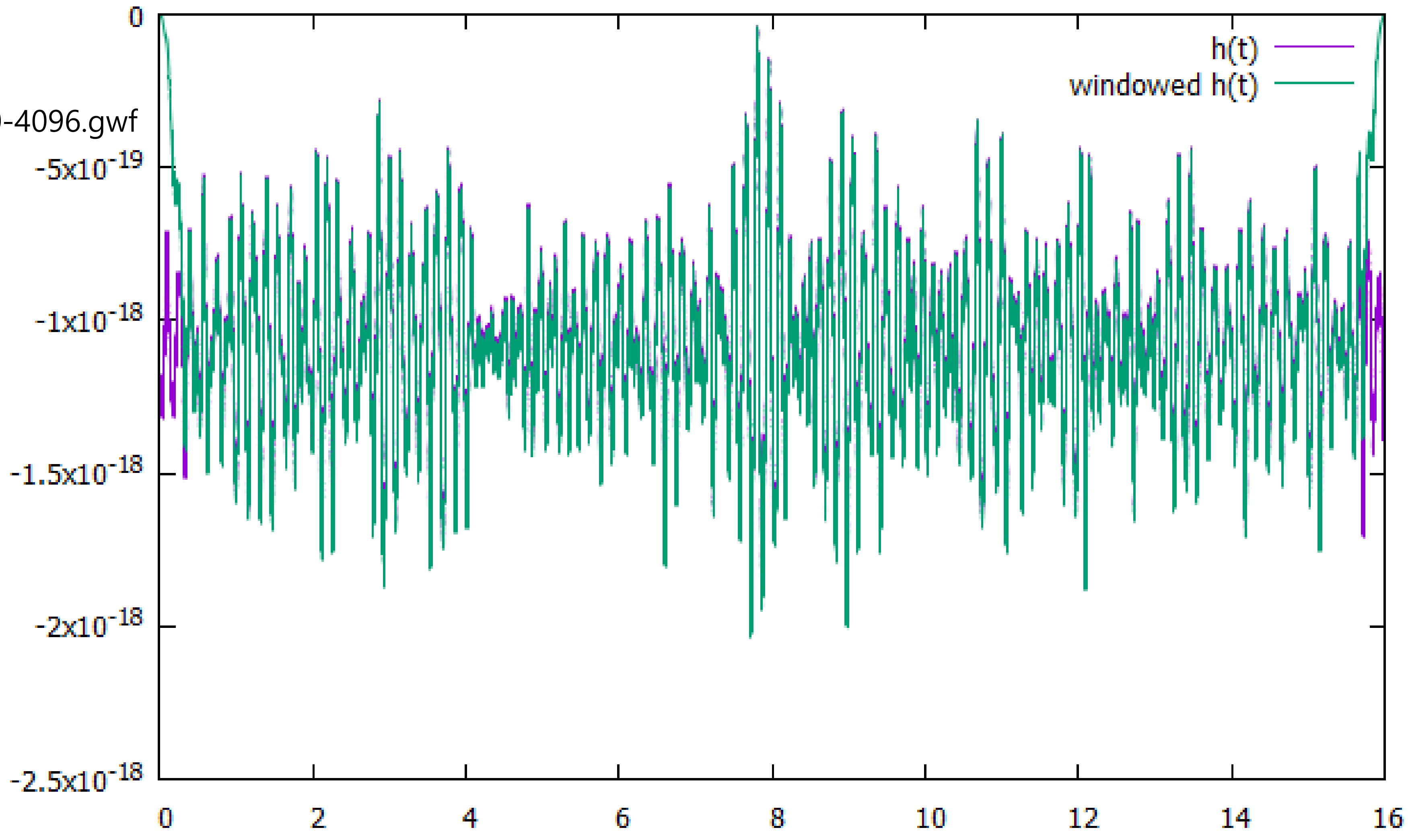
Window function : Tukey with 5% padding

Strain data : L-L1_LOSC_16_V1-1126256640-4096.gwf

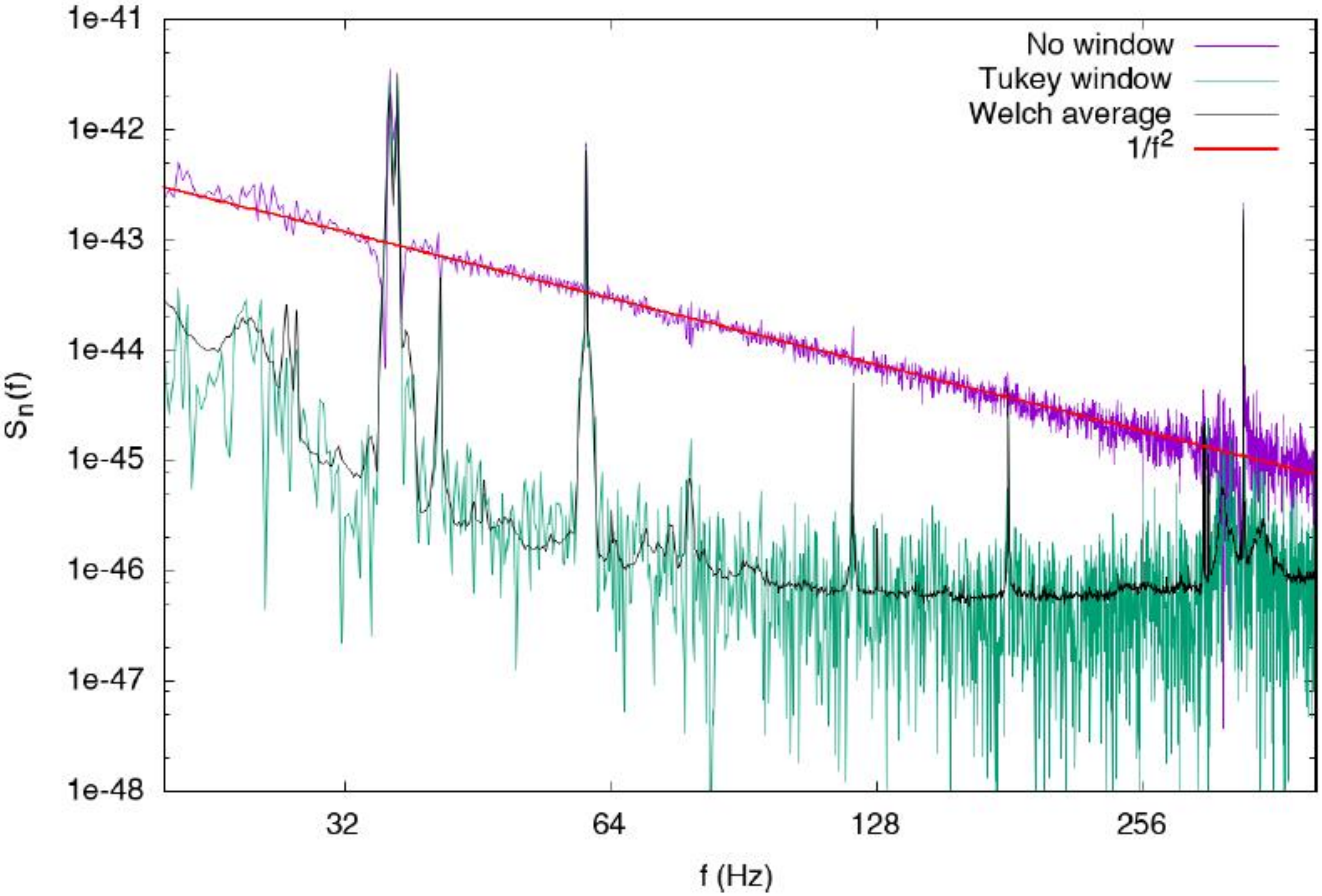
Channel : L1:GWOSC-16KHZ_R1_STRAIN

Trigger time : 1126259462

Segment length : 16s

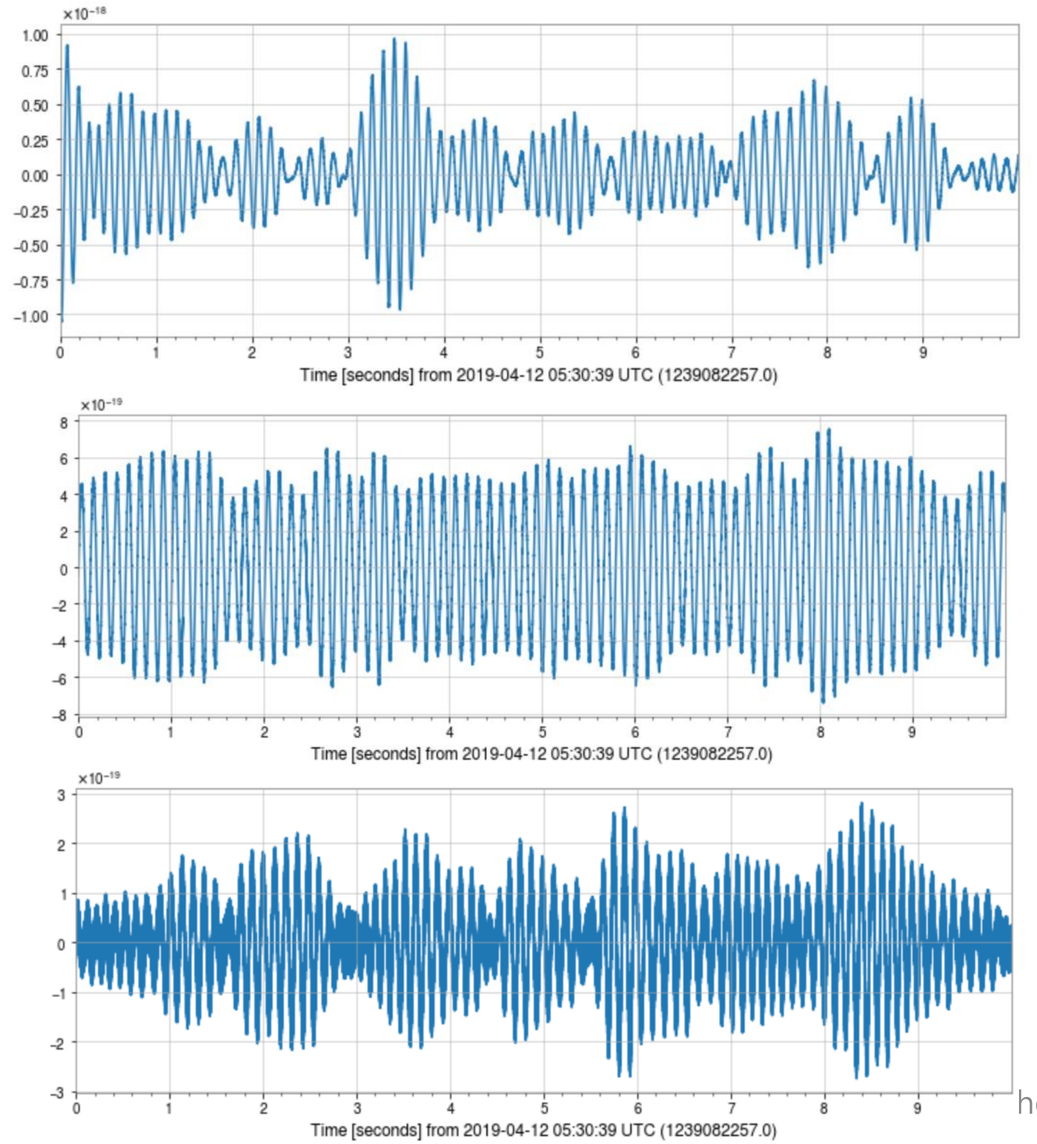


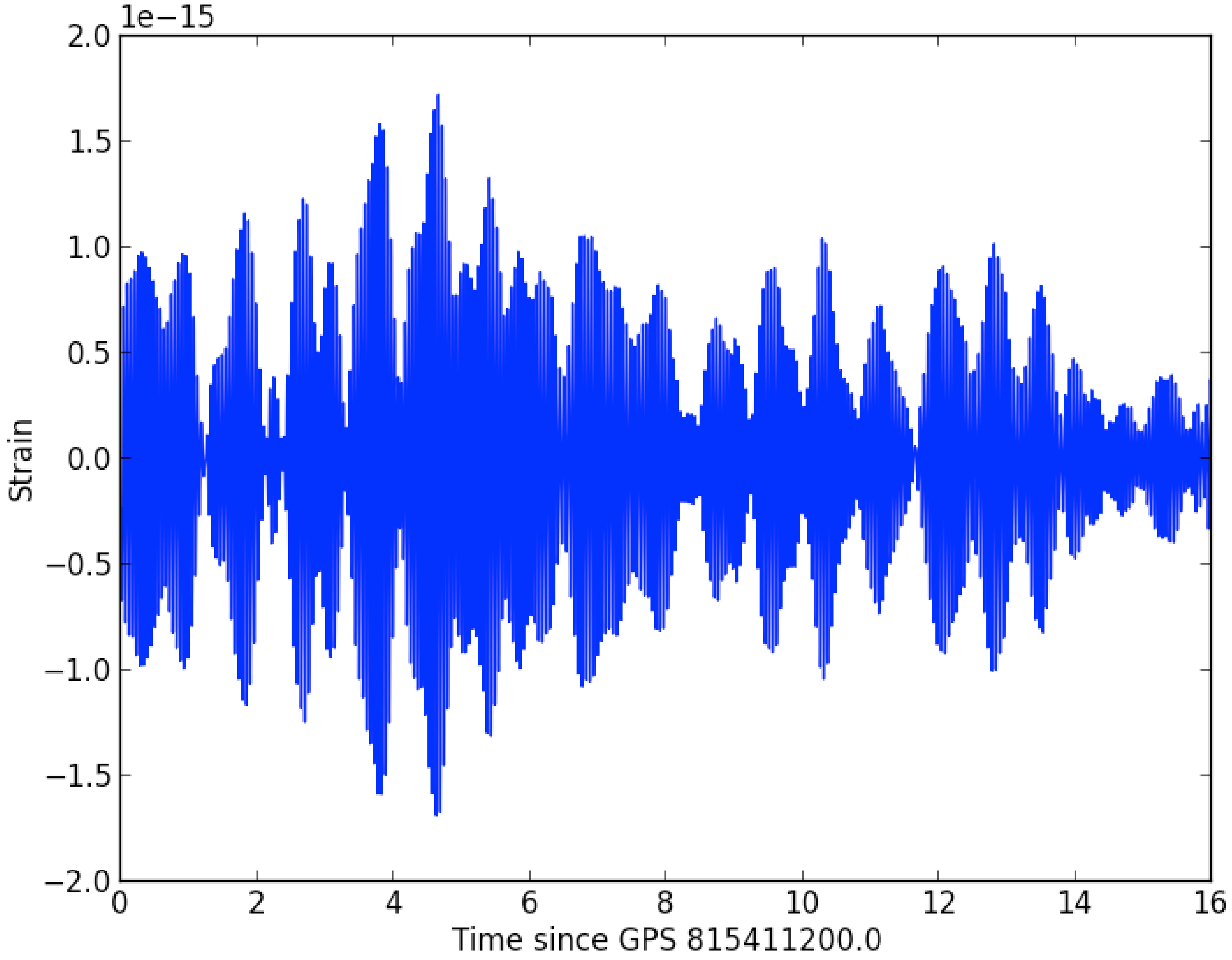
스펙트럼



No window
Tukey window
Welch average
$1/f^2$
$S_n(f)$
f (Hz)
1e-41
1e-42
1e-43
1e-44
1e-45
1e-46
1e-47
1e-48
32
64
128
256

LIGO 자료 예



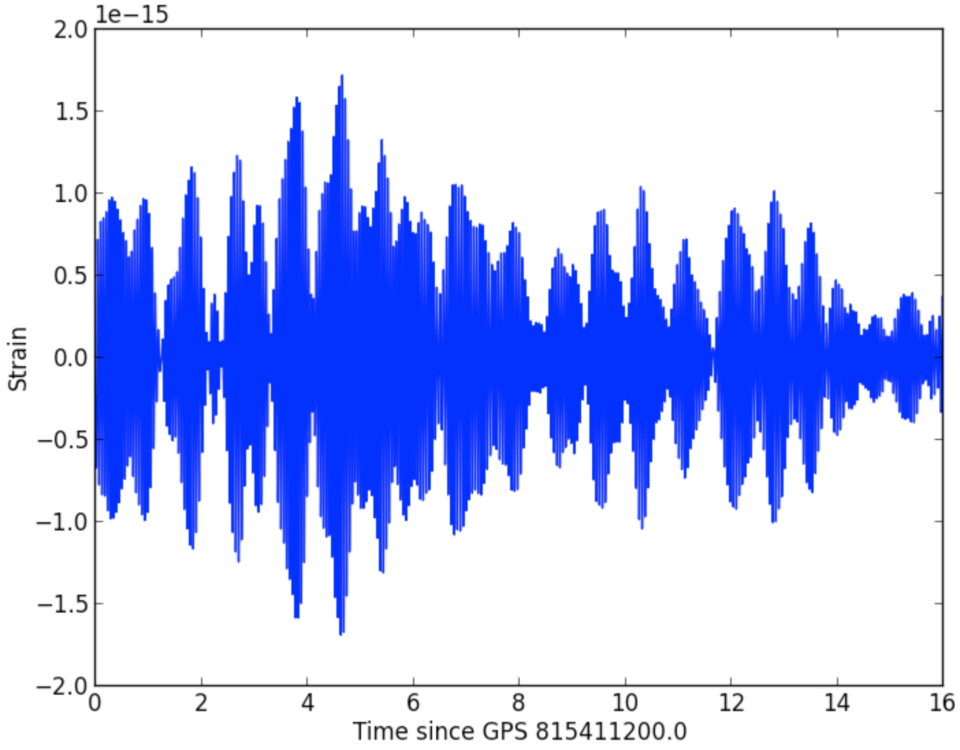


2005-11-07T15:06:27

Odw4, GW190412 [GWOSC (gw-openscience.org)](https://www.gw-openscience.org)

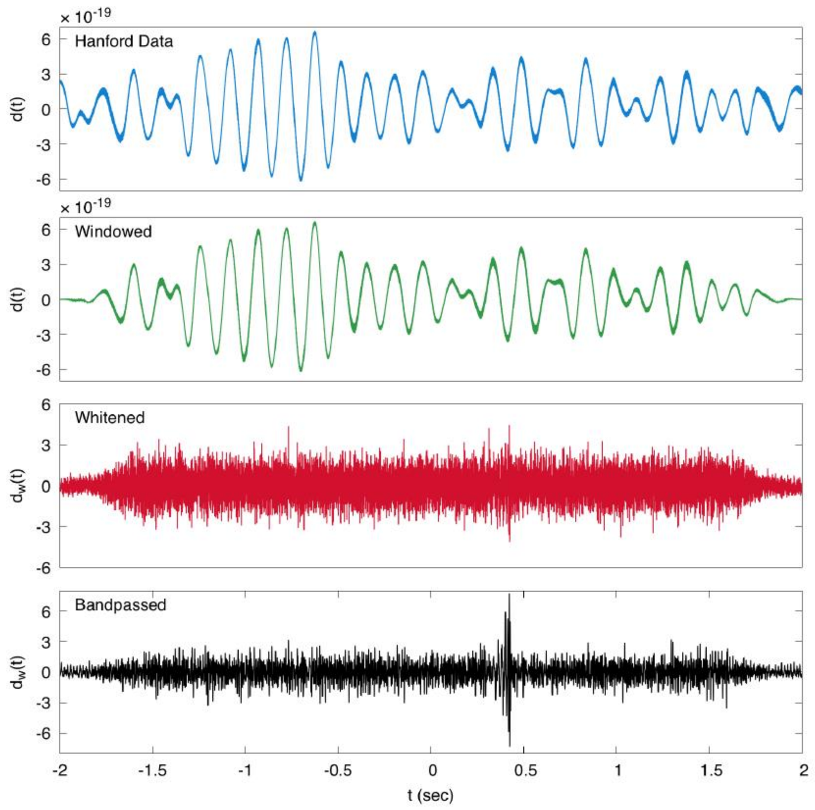
LIGO 자료

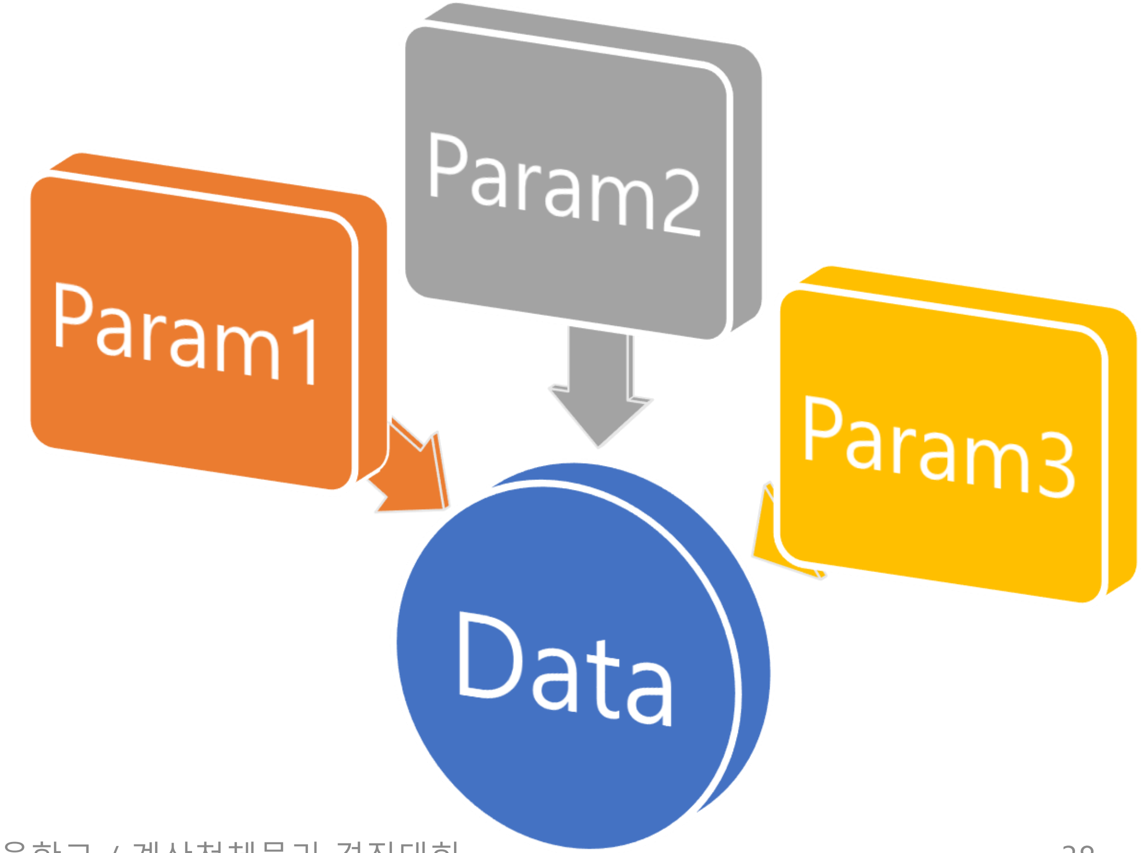
- Discretely sampled time-series data
- Sampling rate (fs)
- h(t) – calibrated strain
  - ALSO: hundreds of "auxiliary" channels
- Recorded at 16384 Hz sample rate
- ~300 MB per hour
- Stored in .gwf "frame" files
  - Also HDF5(Hierarchical Data Format version 5)



베이지언 추론

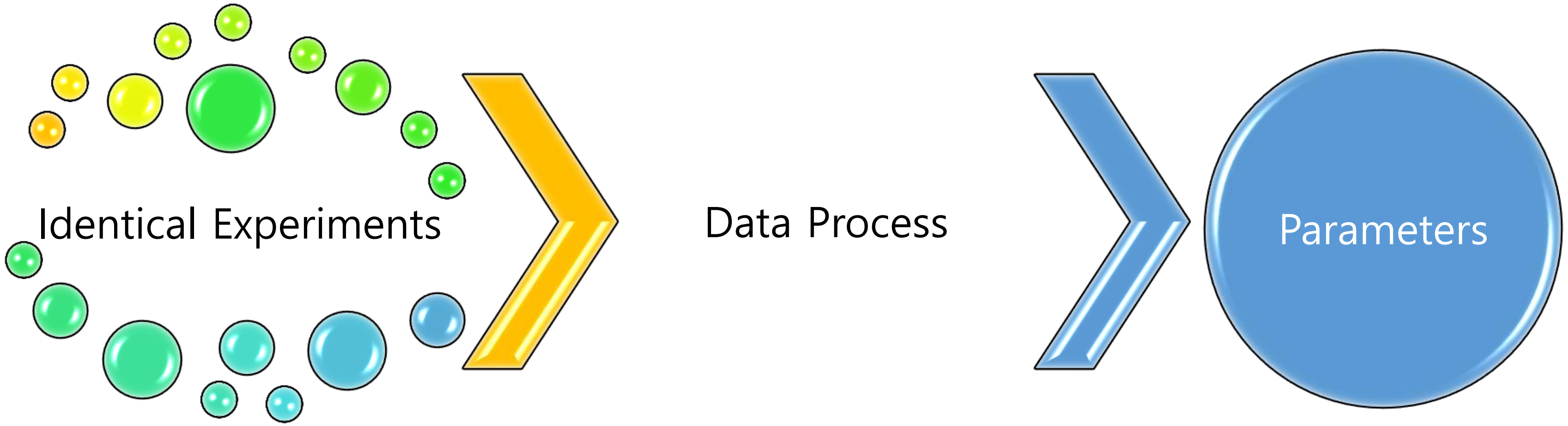
- 질문: “관측된 자료를 가장 잘 설명하는 중력파원의 물리적 매개변수의 분포는 어떻게 되는가?”

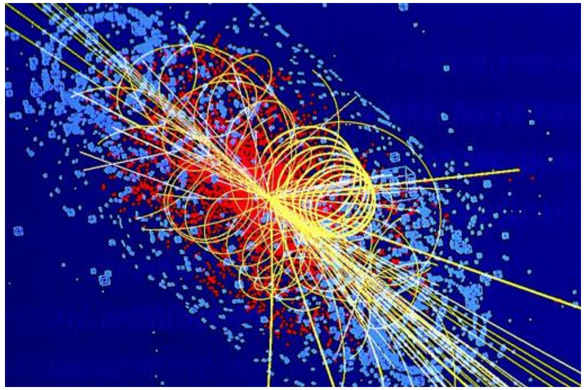


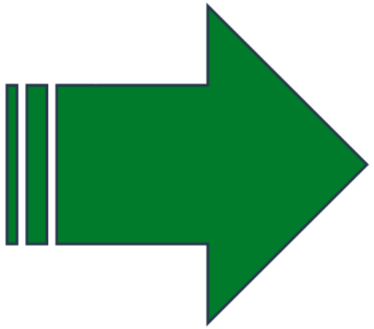


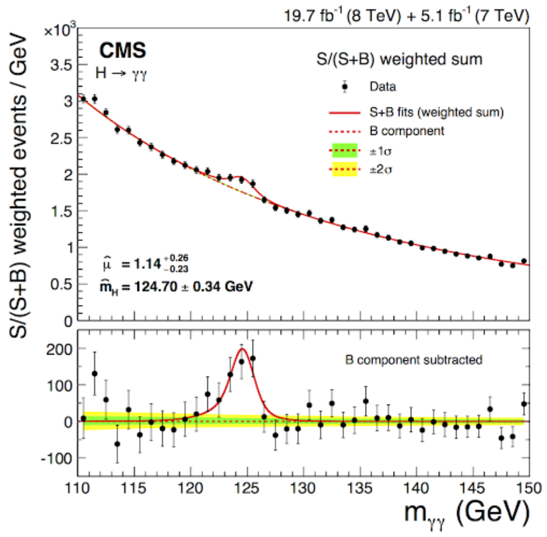
실험실 자료 처리

- 질문 : "동일한 조건의 많은 실험을 통하여 획득한 물리적인 매개변수의 분포는 어떻게 되는가?"









베이지언 추론 기본

- Bayes' rule(theorem)

사후확률

가능도 함수

사전확률

결합확률

$$p(M|D) = \frac{p(D|M)\, p(M)}{p(D)} \qquad p(M,D) = p(M|D)p(D) = p(D|M)p(M)$$

- 모델에 대한 개선된 믿음의 정도는 기존 믿음의 정도와 모델이 관측된 자료를 생성할 확률의 곱이다.

$$p(M, \boldsymbol{\theta}|D, I) = \frac{p(D|M, \boldsymbol{\theta}, I)\, p(M, \boldsymbol{\theta}|I)}{p(D|I)}$$

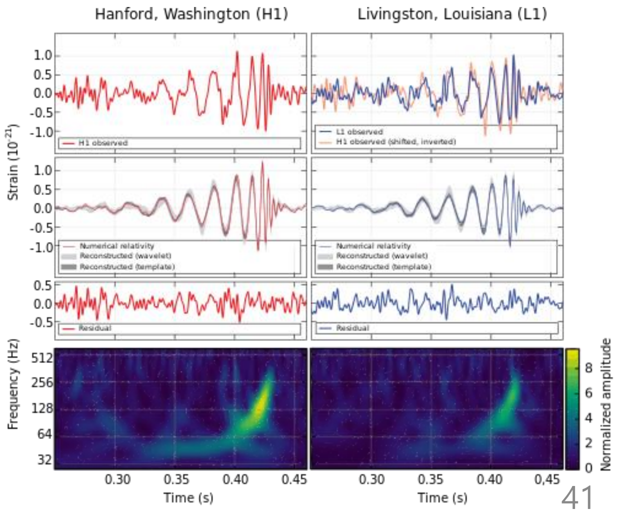
$$p(M, \boldsymbol{\theta}|I) = p(\boldsymbol{\theta}|M, I)\, p(M|I)$$

가능도 함수(Likelihood Function)

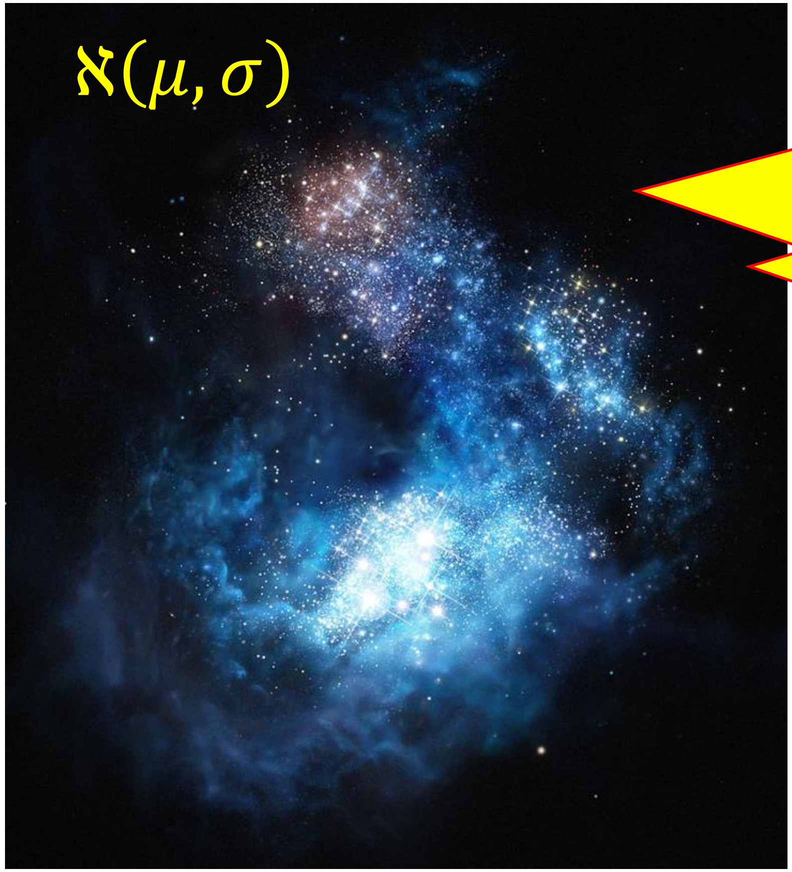
- 관측된 자료에 대한 가능도 함수는 실험장치에 대한 정량적 지표이다.

Understand Device → Calculate Probability

Likelihood

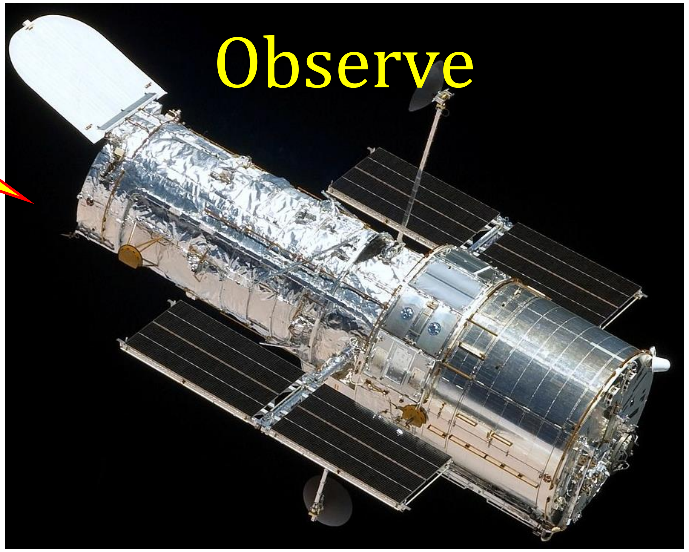


가능도 함수



The probability of the data given the model

$$L \equiv p(\{x_i\}|M(\boldsymbol{\theta})) = \prod_{i=1}^{n} p(x_i|M(\boldsymbol{\theta}))$$



$$x_i: \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

$$\prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)$$

Likelihood is a function of x for given model

Likelihood is a function of model parameters for given data

내적(Inner Product)

$$K(f) \propto \frac{h(f)}{S_n(f)}$$

- $\langle a|b\rangle \equiv 4\Re \int_0^{\infty} \frac{\tilde{a}(f)\tilde{b}^*(f)}{S(f)} df$
- $\langle a|b\rangle = 2\Re \int_{-\infty}^{\infty} \frac{\tilde{a}(f)\tilde{b}^*(f)}{S(|f|)} df = \int_{-\infty}^{\infty} \frac{\tilde{a}(f)\tilde{b}^*(f)+\tilde{a}^*(f)\tilde{b}(f)}{S(|f|)} df$
- $p_x[x(t)] \propto e^{-\langle x|x\rangle/2}$

- We can consider each frequency bin distributed as
- $p_x[\tilde{x}(f)] = \frac{1}{\sqrt{2\pi}}\sqrt{\frac{4\Delta f}{S_x(f)}}\, e^{-\frac{1}{2}\frac{|\tilde{x}(f)|^2 \Delta f}{S_x(f)/4}}$

$$\Delta f = \frac{1}{T}$$

$$p_x[x(t)] \propto \exp\left\{-\frac{1}{2}4\int_0^{\infty}\frac{|\tilde{x}(f)|^2}{S_x}df\right\}$$

가능도 함수(Likelihood)

- $p(d|\theta)$: 모델 매개변수 θ 가 만든 중력파가 신호 d를 생성할 확률
- 신호 d가 파형 $h(\theta)$를 포함할 확률
  - ***만약 $\mathbf{d}$가 파형 $\mathbf{h(\theta)}$를 포함한다면, $\mathbf{d-h}$는 순 잡음이다.***
- $p(d-h|\theta) = p(n|\theta)$
- $p_d\left[\tilde{d}(f_j) - \tilde{h}(f_j)\right] = \frac{1}{\sqrt{2\pi}}\sqrt{\frac{4\Delta f}{S_n(f_j)}}\, e^{-\frac{1}{2}\frac{\left|\tilde{d}(f_j)-\tilde{h}(f_j)\right|^2 \Delta f}{S_n(f_j)/4}}$
- $p(d|\theta) = \prod_{j=0}^{N-1} \frac{1}{\sqrt{2\pi}}\sqrt{\frac{4\Delta f}{S_n(f_j)}}\, e^{-\frac{1}{2}\frac{\left|\tilde{d}(f_j)-\tilde{h}(f_j)\right|^2 \Delta f}{S_n(f_j)/4}} \propto e^{-\langle d-h|d-h\rangle/2}$

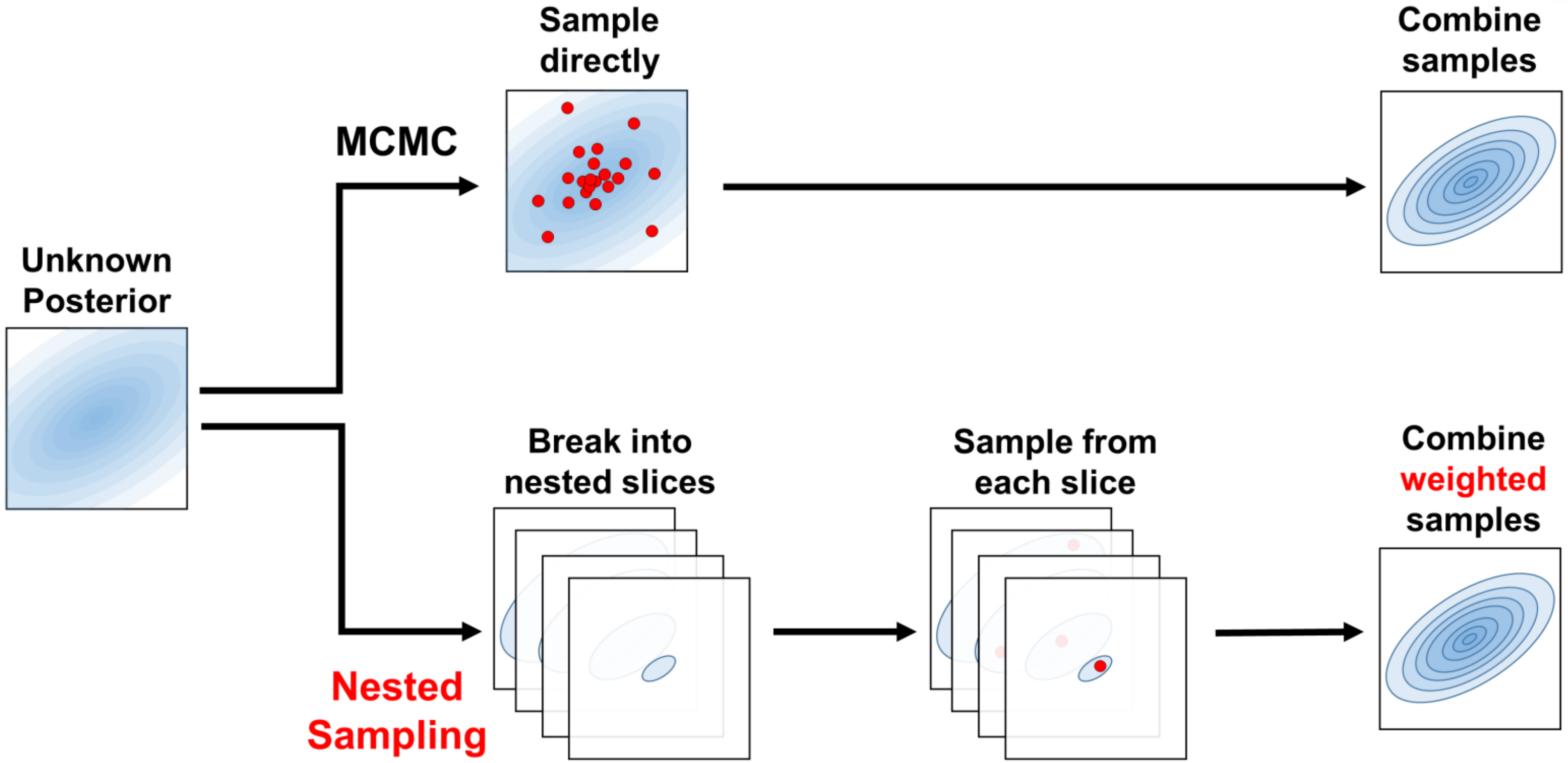
베이지언 추론 과정

1. 가능도 함수 $p(D|\vec{\theta}, M)$ 결정
2. 매개변수에 대한 사전확률 함수 $p(\vec{\theta}|M)$ 결정
3. 사후 확률함수밀도 $p(\vec{\theta}|D, M)$ 계산
4. 최대 사후확률 값(MAP)
   최대 가능도 함수 값(ML)
   사후확률평균 $\bar{\theta} = \int \theta p(\theta|D, M) d\theta$
   marginalization
   $p(\theta|D, M) = \int p\left(\vec{\theta}\middle|D, M\right) d\vec{\theta'}$, 다른 매개변수에 대한 적분
5. 매개변수에 대한 분산
6. 모델과 매개변수에 대한 가설검증

샘플 생성 방법

사후분포를 따르는 많은 독립적인 샘플을 생성한다

- Markov Chain Monte Carlo(MCMC)
- Nested Sampling



$$Z = \int p(M|D)dM$$

$$p(M|D) = \frac{p(D|M)\,p(M)}{p(D)}$$

MNRAS 493, 3132(2020)

MCMC 알고리즘

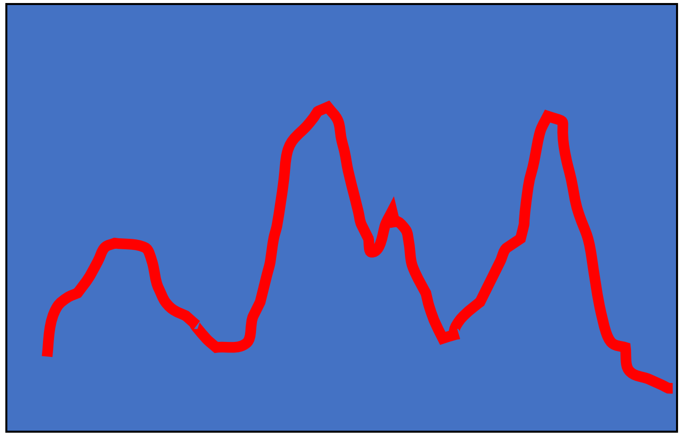
- Metropolis algorithm
- Metropolis-Hasting algorithm
- Gibbs sampling algorithm
- Hamiltonian Monte Carlo
- …

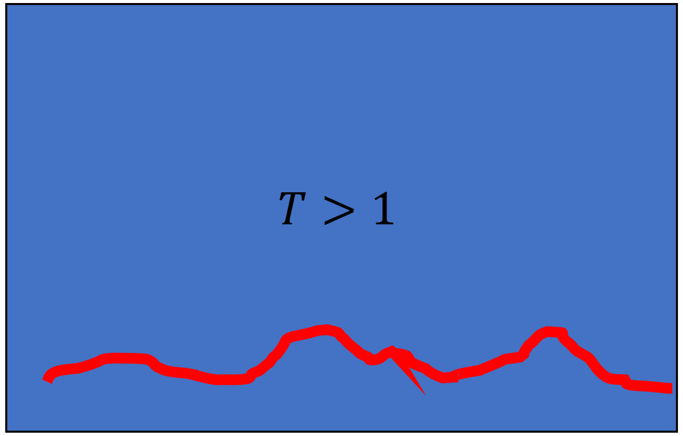
- 초기수렴(Burn In)
- 수렴(Convergence)
- 혼합(Mixing)

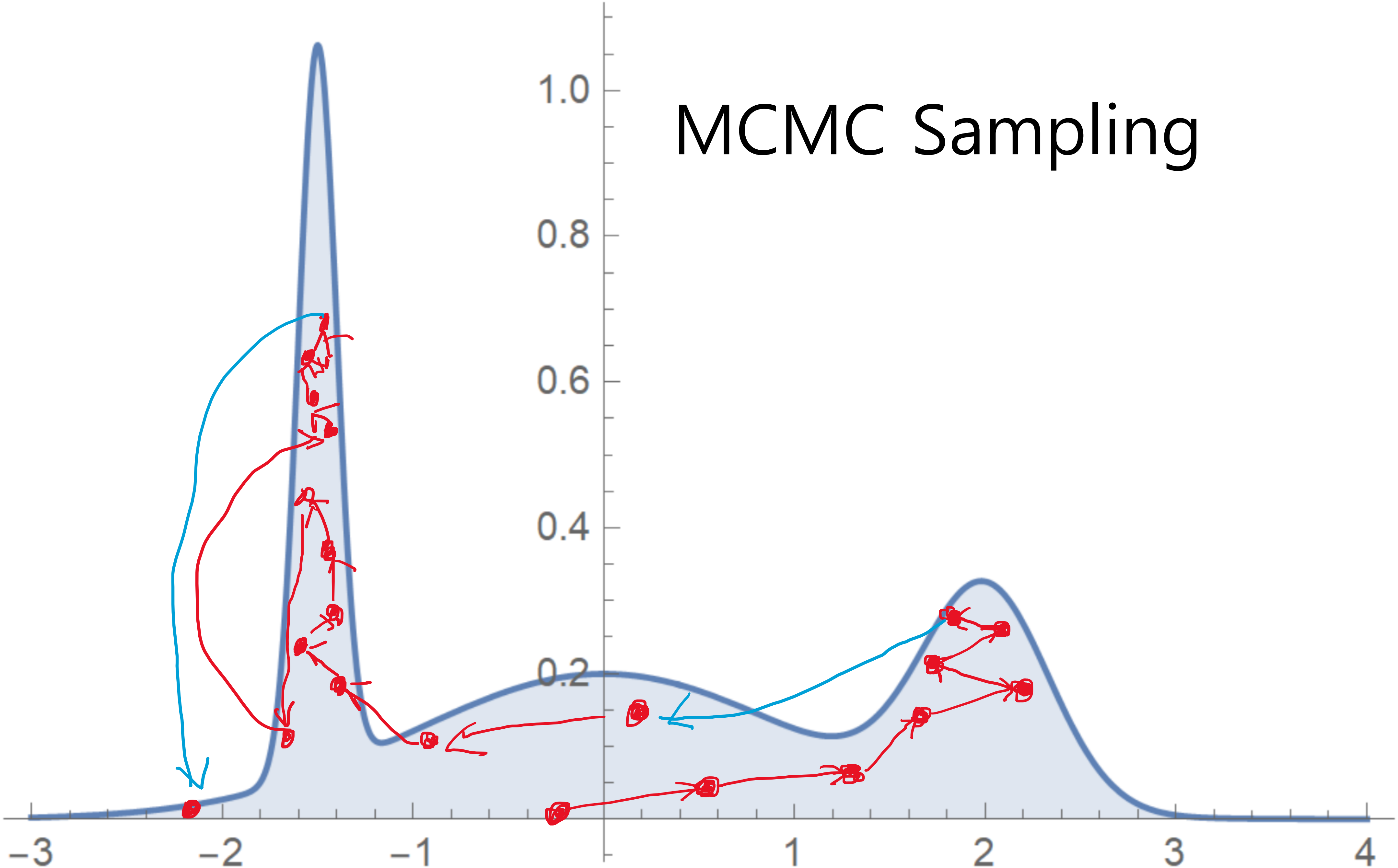
샘플 생성함수에
따라 수렴정도가
달라진다

온도 병렬계산(parallel tempering)

- 몇 개의 다른 가상의 온도를 사용한다
- 가능도 함수를 온도로 조정한다 $p(d|\theta)^{\frac{1}{T}},\ \ T > 1$
- $T_{max} = \frac{(Network\ SNR)^2}{n_{par}}$
- 수렴과 혼합을 개선한다






MCMC Sampling
1.0
0.8
0.6
0.4
0.2
−3
−2
−1
1
2
3
4

Nested Algorithm

- 믿음(evidence) 계산
- $Z = \int_{\Omega_{\vec{\theta}}} p\left(D\middle|\vec{\theta}, M\right) p\left(\vec{\theta}\middle|M\right) d\vec{\theta} = \int_0^1 p(D, M, X)dX$
- $X(\lambda) = \int_{\Omega_{\vec{\theta}}: p\left(D\middle|\vec{\theta}, M\right) \geq \lambda} p\left(\vec{\theta}\middle|M\right) d\vec{\theta}\,, X(\lambda = 0) = 1, X(\lambda = \infty) = 0$

높은 차원의 적분을 효과적으로 1차원 적분으로 계산

Algorithm 1: Static Nested Sampling

```
// Initialize live points.
```
Draw K "live" points$\{\mathbf{\Theta}_1, \ldots, \mathbf{\Theta}_K\}$ from the prior $\pi(\mathbf{\Theta})$.
```
// Main sampling loop.
```
while stopping criterion not met **do**

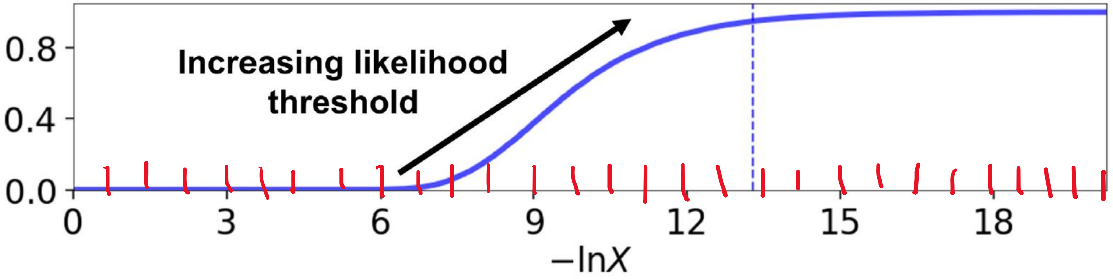
  Compute the minimum likelihood $\mathcal{L}^{\min}$ among thecurrent set of live points.
  Add the kth live point $\mathbf{\Theta}_k$ associated with $\mathcal{L}^{\min}$to a list of "dead" points.
  Sample a new point $\mathbf{\Theta}'$ from the prior subject to the constraint$\mathcal{L}(\mathbf{\Theta}') \geq \mathcal{L}^{\min}$.
  Replace $\mathbf{\Theta}_k$ with $\mathbf{\Theta}'$.
  ```
  // Check whether to stop.
  ```
  Evaluate stopping criterion.

end
```
// Add final live points.
```
while $K > 0$ **do**

  Compute the minimum likelihood $\mathcal{L}^{\min}$ among thecurrent set of live points.
  Add the kth live point $\mathbf{\Theta}_k$ associated with $\mathcal{L}^{\min}$to a list of "dead" points.
  Remove $\mathbf{\Theta}_k$ from the set of live points.
  Set $K = K - 1$.

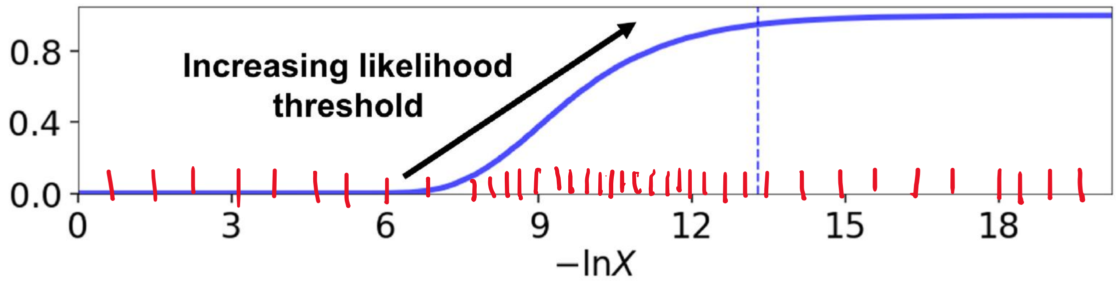
end

등간격 적분



MNRAS 493, 3132(2020)

반응형 간격 적분



Algorithm 2: Dynamic Nested Sampling

```
// Initialize first set of live points.
```
Draw K "live" points$\{\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_K\}$ from the prior $\pi(\boldsymbol{\Theta})$.
```
// Main sampling loop.
```
Set $\mathcal{L}^{\min} = 0$ and $K_0 = K$.

while stopping criterion not met **do**

    `// Get current number of live points.`

    Compute the previous number of live points K and thecurrent number of live points K'.

    if $K' \geq K$ **then**

        `// Add in new live points.`

        while $K' > K$ **do**

            Sample a new point $\boldsymbol{\Theta}'$ from the prior subject to the constraint$\mathcal{L}(\boldsymbol{\Theta}') \geq \mathcal{L}^{\min}$.

            Add $\boldsymbol{\Theta}'$ to the set of live points.

            Set $K = K + 1$.

        end

        `// Replace worst live point.`

        Compute the minimum likelihood $\mathcal{L}^{\min}$ among thecurrent set of K live points.

        Add the kth live point $\boldsymbol{\Theta}_k$associated with $\mathcal{L}^{\min}$to a list of "dead" points.

        Replace $\boldsymbol{\Theta}_k$ with $\boldsymbol{\Theta}'$.

    else

        `// Iteratively remove live points.`

        while $K' < K$ **do**

            Compute the minimum likelihood $\mathcal{L}^{\min}$ among thecurrent set of $K = K'$ live points.

            Add the kth live point $\boldsymbol{\Theta}_k$ associated with $\mathcal{L}^{\min}$to a list of "dead" points.

            Remove $\boldsymbol{\Theta}_k$ from the set of live points.

            Set $K = K - 1$.

        end

    end

    `// Check whether to stop.`

    Evaluate stopping criterion.

end

`// Add final live points.`

while there are live points remaining **do**

    Compute the minimum likelihood $\mathcal{L}^{\min}$ among thecurrent set of live points.

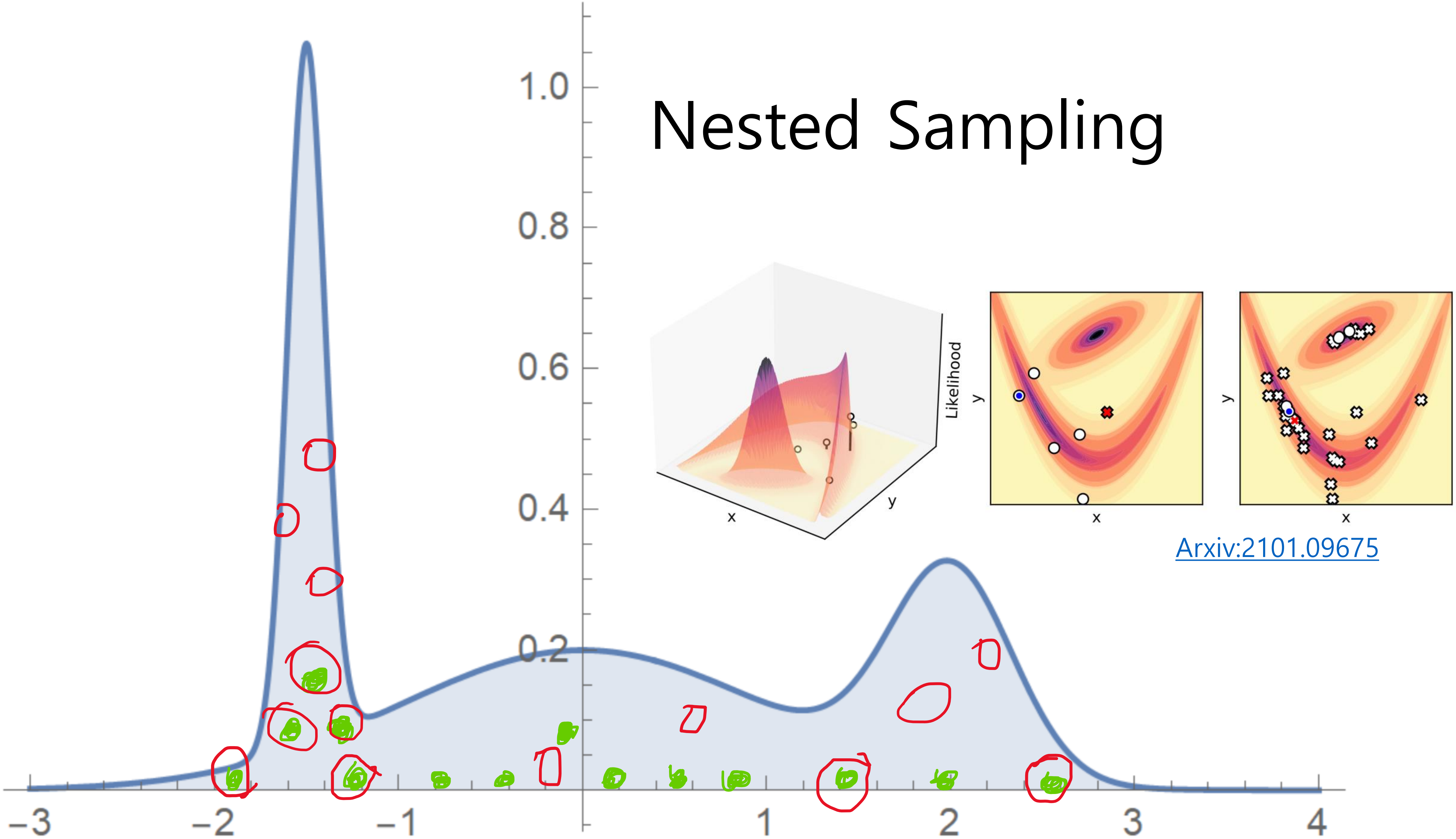
    Add the kth live point $\boldsymbol{\Theta}_k$ associated with $\mathcal{L}^{\min}$to a list of "dead" points.

    Remove $\boldsymbol{\Theta}_k$ from the set of live points.

end

MNRAS 493, 3132(2020)

Nested Sampling



Arxiv:2101.09675

분석도구

LALSuite(LALSuite: Main Page (ligo.org))

- C based LSC Algorithm Library Suite
- LALInferenceMCMC.c : MCMC Sampler
- LALInferenceNest.c : Nested Sampler

- RIFT(Rapid Iterative Fitting)

LALSuite from source

Building LALSuite from source

Dependencies

Build tools

The following build tools will be needed to build LALSuite components from source

- a C compiler with support for the C99 standard
- autoconf (building from git only)
- automake (building from git only)
- make
- pkg-config

Library dependencies

For LAL the library dependencies are:

- GSL - The GNU Scientific Library.
- FFTW - The Fastest Fourier Transform in the West.
- HDF5 - The HDF5 library
- zLib - A Massively Spiffy Yet Delicately Unobtrusive Compression Library

Other subpackages need at least (but not limited to):

- !FrameL- LIGO/Virgo Frame library (needed for LALFrame)
- !MetaIO - LIGO_LW XML library (LALMetaIO)
- CFITSIO - A FITS File Subroutine Library (needed for LALPulsar)

All Dependencies can be installed using an appropriate Package manager, **you should not need to compile any of these yourself**.

The Python layers for each subpackage will have extra requirements that are not specified here.

LALSuite from source

Building from the git repository

The repository is hosted on the LIGO GitLab instance, please see the following computing guide page for details on accessing repositories hosted here. The LALSuite repository also utilizes `git-lfs` for the management of large file so please ensure that you have configured `git-lfs` on your system.

You can then clone the repository using:

```
git clone git@git.ligo.org:lscsoft/lalsuite.git
```

You can also clone using the https interface but the above SSH URL is recommended as this is more rebust:

```
git clone https://git.ligo.org/lscsoft/lalsuite.git
```

If you are cloning anonymously then you *must* use the https URL.

You can then install LAL as follows:

```
LAL_INSTALL_PREFIX="${HOME}/opt/lalsuite"  # change as appropriate
pushd lal
./00boot
./configure --prefix=${LAL_INSTALL_PREFIX}
make
make install
```

LALSuite install in conda

Cloning the Repository

We now utilize Git LFS for the managament of large files and as such `git-lfs` needs to be installed and configured to correctly clone this repository. After installing `git-lfs` it can be configured using:

```
$ git lfs install
```

This only needs to be done once for each machine you access the repository. It can then be cloned using:

```
$ git clone git@git.ligo.org:lscsoft/lalsuite.git
```

Building from Source

The recommended way to build LALSuite from source is in a `conda` environment. A recipe file is available with all main dependencies. This can serve as the base for custom recipes, or be used directly via:

```
$ conda env create -f conda/environment.yml
```

Pulling in dependencies may take a while depending on your internet connection. After the environment setup succeeded, you can activate it with:

```
$ conda activate lalsuite-dev
```

You can then build the suite by executing, in order:

1. `./00boot` (once at first time)
2. `./configure` with appropriate options (see `./configure --help`)
3. `make`

After pulling updates or making your own changes, you will usually only need to call `make` again, as reconfiguration and re-running `00boot` should be handled automatically if needed.

LALInference(LALInference: Main Page (ligo.org))

lalinference_mcmc
lalinference_nest

Documentation

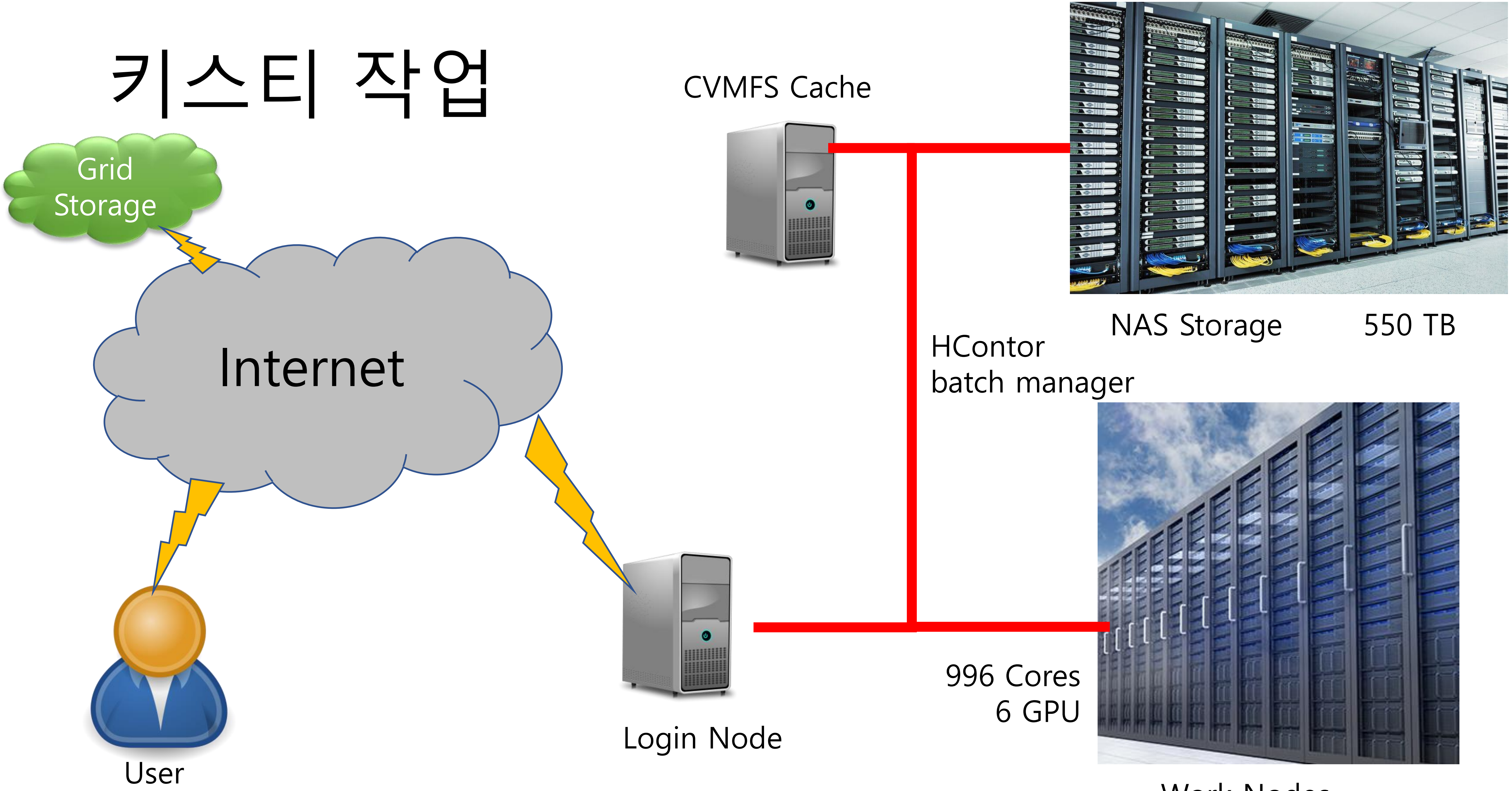
Here is a list of all modules:

▼ General Packages	
Header LALInference.h	Main header file for LALInference common routines and structures
Header LALInferenceLikelihood.h	Header file for likelihood functions used by LALInference codes
Header LALInferenceNestedSampler.h	Nested sampler written for LALInference
Header LALInferencePrior.h	Collection of commonly used Prior functions and utilities
Header LALInferenceProposal.h	Jump proposals for exploring the GW signal parameter space
Header LALInferenceReadData.h	Utility functions for handling IFO data
Header LALInferenceRemoveLines.h	Utility functions for identifying lines in IFO data to be removed in LALInference
Header LALInferenceTemplate.h	Main header file for LALInference signal template generating functions
Header LALInferenceVCSInfo.h	VCS and build information for LALInference
▼ SWIG Interfaces	This package provides Octave and Python wrappings of LALInference functions and data structures generated using SWIG
Interface SWIGLALInferenceAlpha.i	SWIG code which must appear *before* the LALInference headers
Interface SWIGLALInferenceOmega.i	SWIG code which must appear *after* the LALInference headers
Python Packages	

SWIG(Simplified Wrapper and Interface Generator)

- SWIG is a software development tool that connects programs written in C and C++ with a variety of high-level programming languages. SWIG is used with different types of target languages including common scripting languages such as Javascript, Perl, PHP, Python, Tcl and Ruby. The list of supported languages also includes non-scripting languages such as C#, D, Go language, Java including Android, Lua, OCaml, Octave, Scilab and R. Also several interpreted and compiled Scheme implementations (Guile, MzScheme/Racket) are supported. SWIG is most commonly used to create high-level interpreted or compiled programming environments, user interfaces, and as a tool for testing and prototyping C/C++ software. SWIG is typically used to parse C/C++ interfaces and generate the 'glue code' required for the above target languages to call into the C/C++ code. SWIG can also export its parse tree in the form of XML. SWIG is free software and the code that SWIG generates is compatible with both commercial and non-commercial projects.

키스티 작업



CVMFS Cache
Grid Storage
Internet
NAS Storage 550 TB
HContor batch manager
996 Cores
6 GPU
Login Node
User
Work Nodes

작업 수행

[screen 0: bash] hwlee@ldg-ui01:/data/ligo/scratch/hwlee/eccwork/gw151226/ecc-ecc-0.04

```
#!/bin/sh

module load openmpi-x86_64
#source /opt/intel/parallel_studio_xe_2018.1.038/bin/psxevars.sh
#export masterdir=/data/ligo/scratch/pe/LAL/lalinference_o2_eccTides/
export masterdir=/data/ligo/scratch/pe/LAL/eccentricity_160810
source $masterdir/etc/lscsoftrc
export PYTHON=/usr/lib64/python2.7/site-packages

date

cd /data/ligo/scratch/hwlee/eccwork/gw151226/ecc-ecc-0.04
 condor_submit 0noiseecc0.sub
 condor_submit 0noiseecc0.sub
 condor_submit 0noiseecc0.sub
 condor_submit 0noiseecc0.sub
 condor_submit 0noiseecc0.sub

exit
```

[screen 0: bash] hwlee@ldg-ui01:/data/ligo/scratch/hwlee/eccwork/gw151226/ecc-ecc-0.04

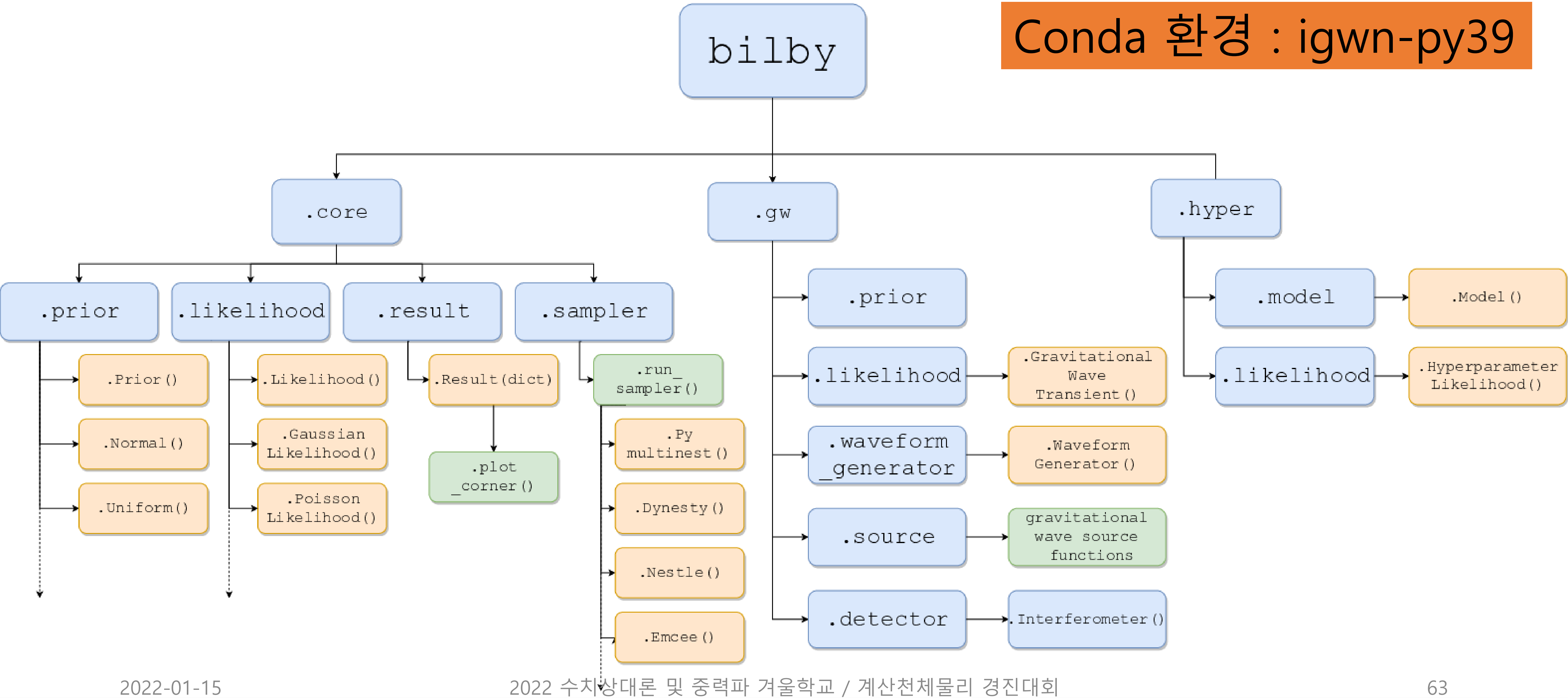
```
universe   = vanilla
getenv     = true
executable = /usr/lib64/openmpi/bin/mpirun
arguments  = -np 11 /data/ligo/scratch/pe/LAL/eccentricity_160810/bin/lalinference_mcmc --outfile PTMCMC.output.$(Cluster)-$(Process)
.h5 --ifo H1 --H1-cache LALSimAdLIGO --H1-flow 25 --trigtime 894383679.0 --psdstart 894383379.0 --psdlength 1024.0  --seglen 16 --sra
te 2048 --inj ./taylorF2EccH1onlyBBHGW151226.xml  --event 4 --inj-fref 100  --inj-spinOrder 0 --inj-tidalOrder -1 --approx TaylorF2Ec
cthreePointFivePN  --fref 100  --nsteps 20000000  --skip 100 --neff 10000   --amporder Newtonian --spinOrder 0 --tidalOrder -1 --radi
ation-frame --margtime --tempLadderBottomUp --differential-buffer-limit 100000  --dataseed 12345 --0noise --noSpin --tidalT --quadpar
am

output     = $(Cluster)-$(Process).f2ecc-ecc.gw151226.0noise.out
error      = $(Cluster)-$(Process).f2ecc-ecc.gw151226.0noise.err
log        = $(Cluster)-$(Process).f2ecc-ecc.gw151226.0noise.log
request_cpus = 11
request_memory = 11*6*1024

queue 1
```

Bilby(Welcome to bilby's documentation! — bilby 1.1 documentation (ligo.org))

Conda 환경 : igwn-py39



Installation

Conda | Pip

```
$ conda install -c conda-forge bilby
```

Supported python versions: 3.6+.

This will install all requirements for running `bilby` for general inference problems, including our default sampler dynesty. Other samplers will need to be installed via pip or the appropriate means.

Dynesty Guide

The Dynesty sampler is just one of the samplers available in bilby, but it is well-used and found to be fast and accurate. Here, we provide a short guide to its implementation. This will not be a complete guide, additional help can be found in the Dynesty documentation.

All of the options discussed herein can be set in the `bilby.run_sampler()` call. For example, to set the number of live points to 1000

```
>>> bilby.run_sampler(likelihood, priors, sampler="dynesty", nlive=1000)
```

Bilby MCMC Guide

Bilby MCMC is a native sampler built directly in `bilby` and described in Ashton & Talbot (2021). Here, we describe how to use it.

Quickstart and output

To use the `bilby_mcmc` sampler, we call

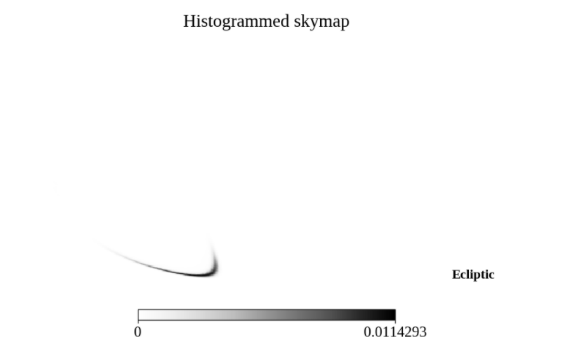
```
>>> bilby.run_sampler(likelihood, priors, sampler="bilby_mcmc", nsamples=1000)
```

This will run the MCMC sampler until 1000 independent samples are drawn from the posterior. As the sampler is running, it will print output like this

후처리 도구

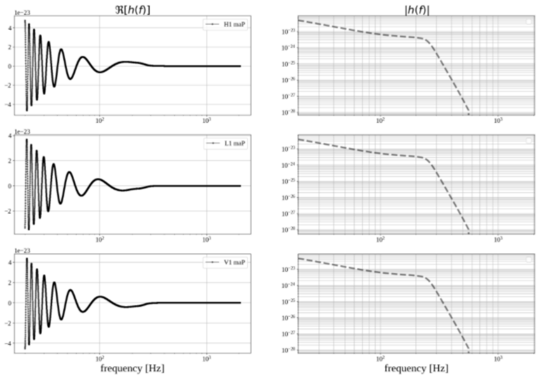
CBCBayesPostProc

SkyMap

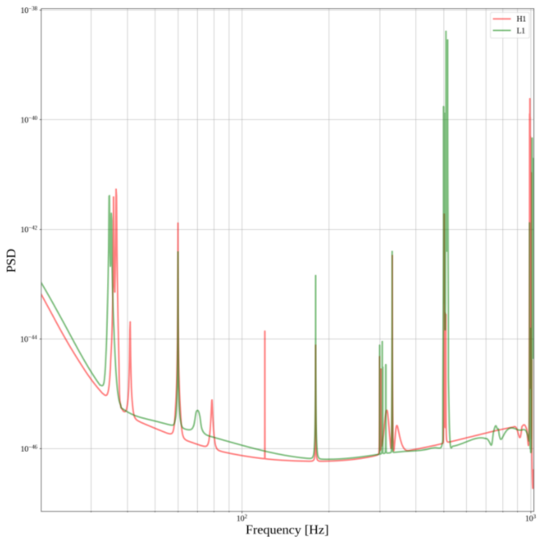


Confidence region	size (sq. deg)
0.67	93.4763
0.9	196.342
0.95	274.344
0.99	666.714

Waveforms

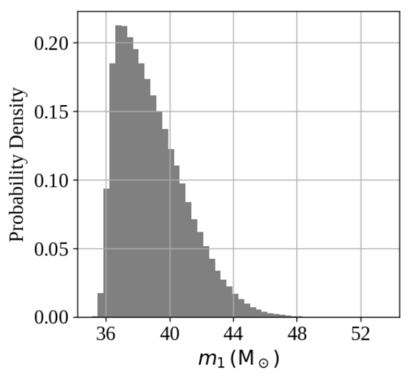


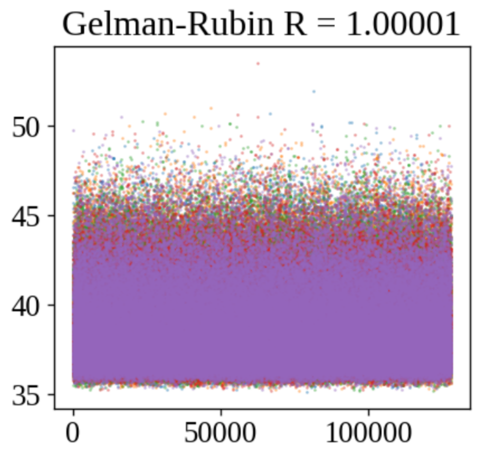
PSDs

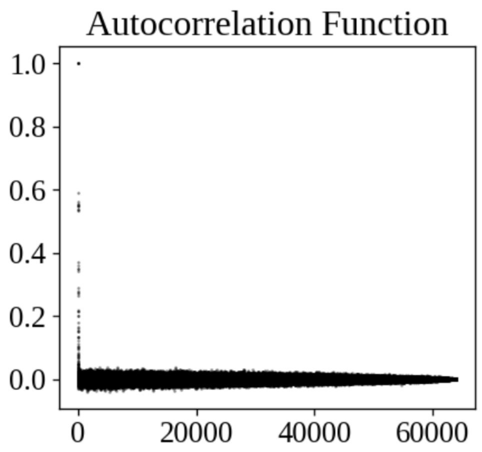


Top	1D marginal posterior PDFs (Masses)	[-] Collapse

Histogram and Kernel Density Estimate	Samples used	Autocorrelation

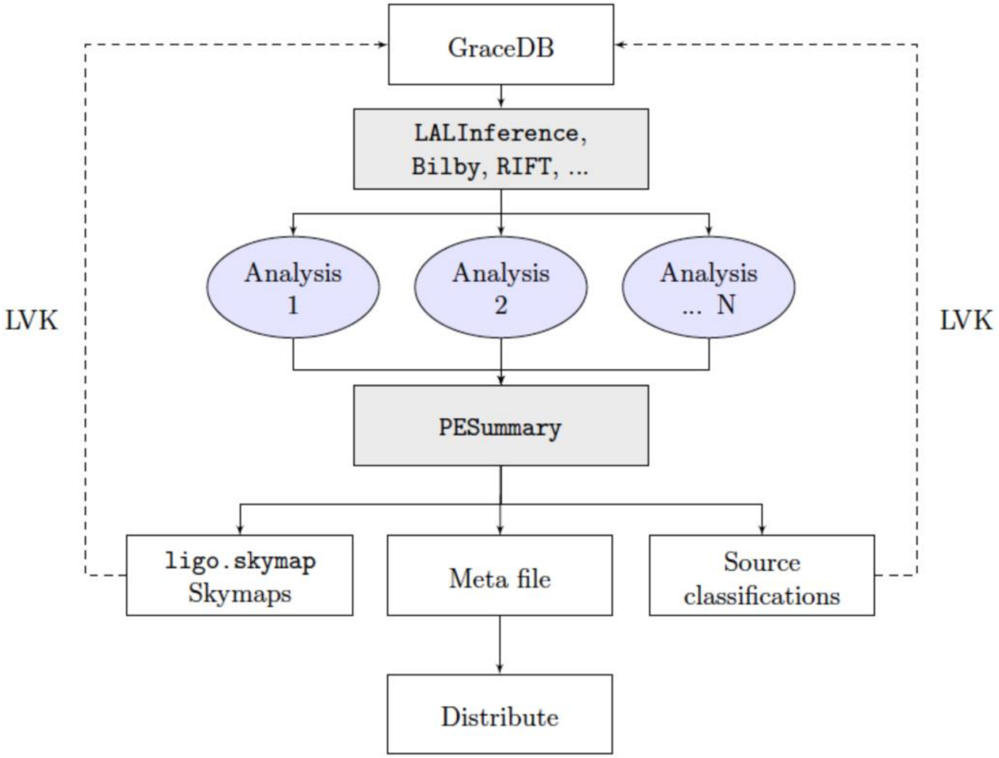






PESummary(PESummary (ligo.org))

- 모수추정 후처리 결과 생성 라이브러리



Install PESummary

`PESummary` is developed and tested for python 3.5+. We recommend that this code is installed inside a virtual environment using `virtualenv`. This environment can be installed with python 3.5+ using pyenv.

For detailed instructions on how to set up your virtual environment, please refer to setting up a virtual environment.

Installing PESummary using pip

If you choose to install `PESummary` using `pip`, then simply run:

```
$ source ~/virtualenvs/pesummary_py3.6/bin/activate
$ pip install pesummary
```

Installing PESummary using conda

If you choose to install `PESummary` using `conda`, then simply run:

```
$ source ~/virtualenvs/pesummary_pyenv3.6/bin/activate
$ conda install -c conda-forge pesummary
```

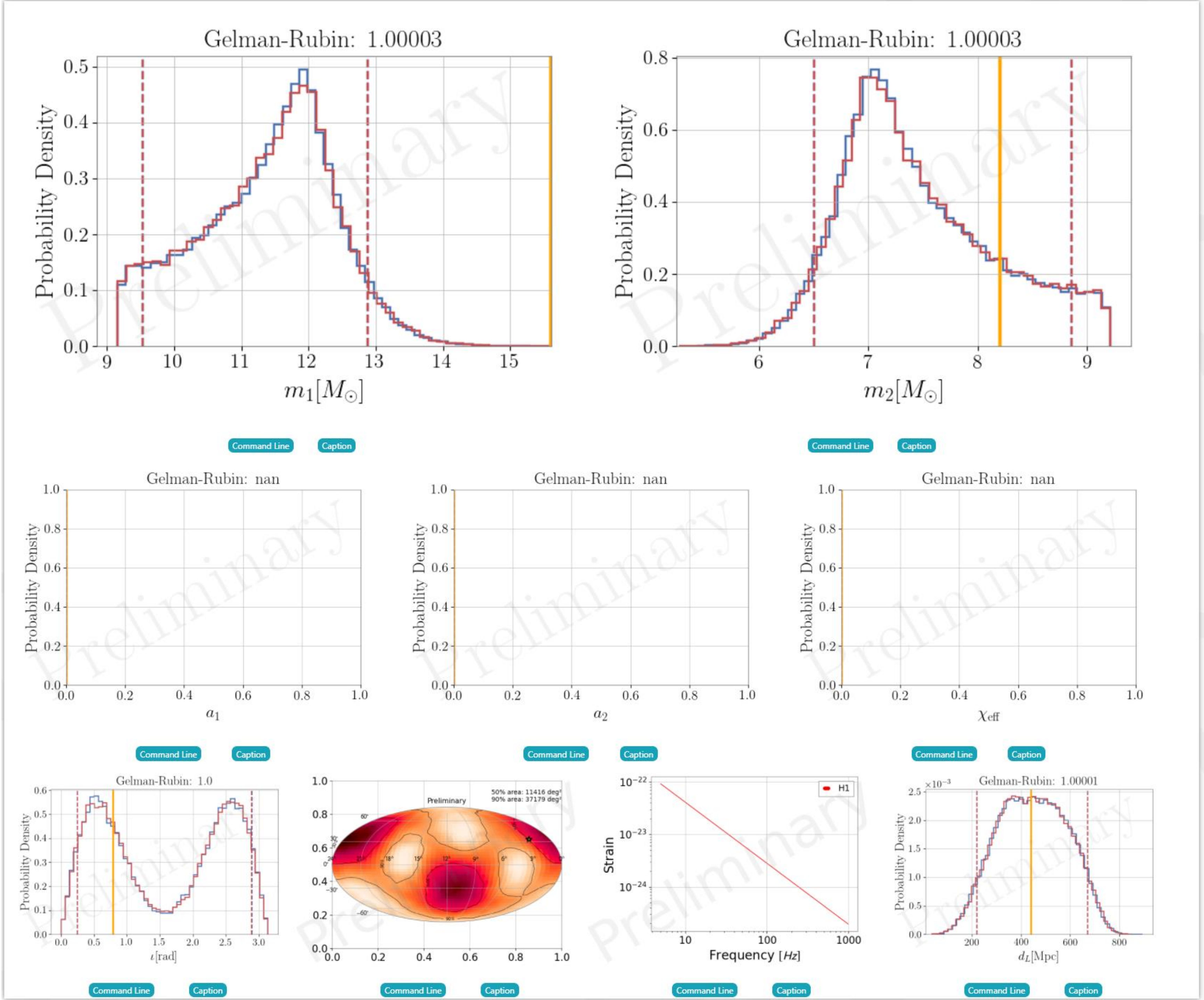
Pulling the PESummary docker image

If you would like, you are able to pull the `PESummary` docker image. To do this, simply run:

```
$ docker pull 08hoyc/pesummary
```

PESummary 출력

- Bootstrap css
- Interactive



맺음 말

결론

중력파 관측은 매우 어려운 일이다

중력파 관측을 통하여 다양한 물리를 할 수 있다

중력파 관측을 위해서는 종합적인 물리, 공학 기술이 필요하다

중력파 물리학의 미래를 위해서 여러분이 필요하다

질문

감사합니다