How to Generate A Gravitational Waveform Template Bank?

Han Wang

June 20th, 2023

Matched Filtering

☐ **Matched filtering:**
  ➢ *A widely used method to detect Gravitational Waves(GWs)*
  ➢ Match between waveform templates and the signal
  ➢ More correlation, more likely to have such waveform in signal
  ➢ Need to use different templates to find the closest one
    ➢ Optimal SNR: $\rho_{\text{opt}}^2 = (h \mid h)$
    ➢ **Need a pre-computed template bank**

• **Inner product:**

$$(h \mid g) \equiv 4\Re \int_0^{+\infty} \frac{\tilde{g}^*(f)\tilde{h}(f)}{S_n(f)} \mathrm{d}f$$
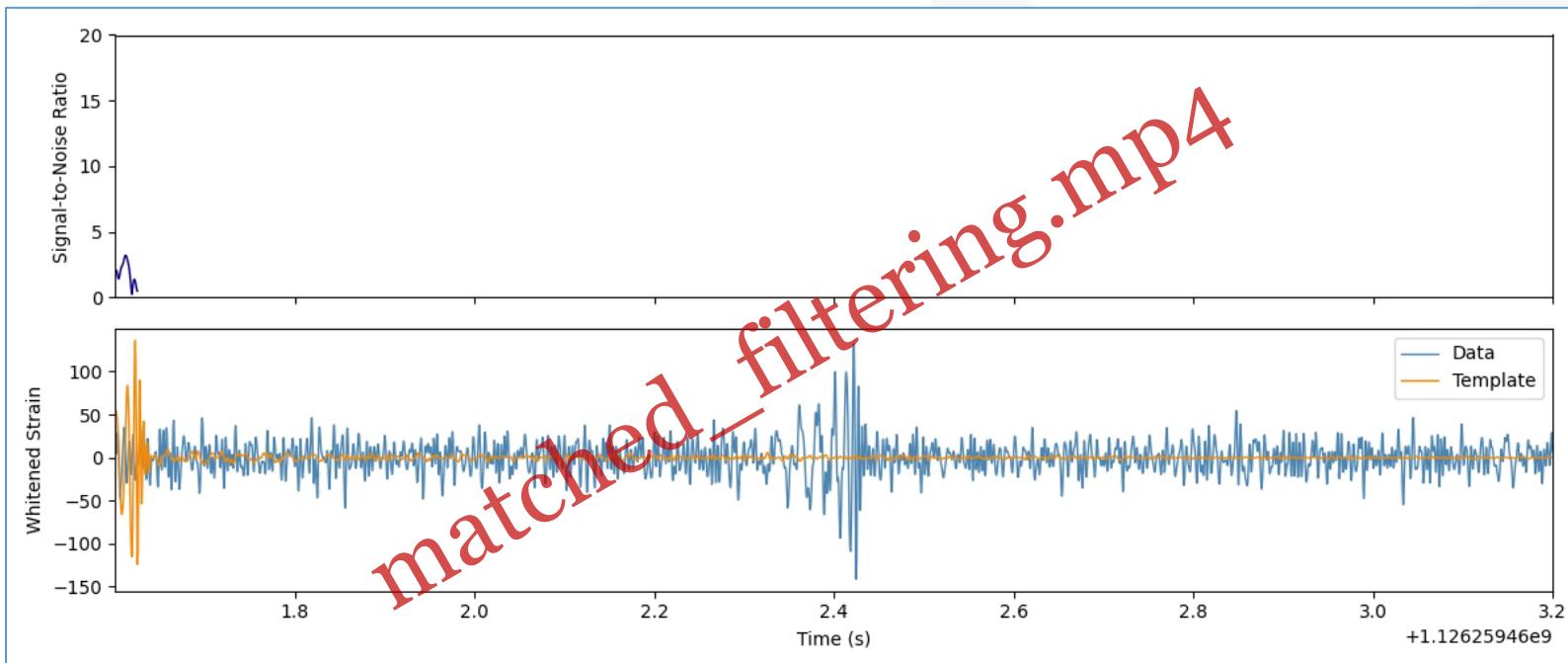

Illustration of matched filtering method (GW150914 as an example)
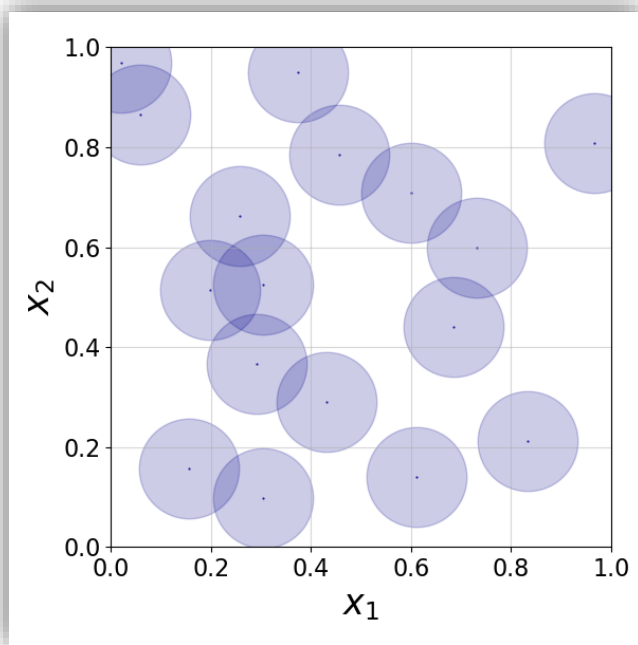
2

Template Bank

☐ **Validity**
  ➤ Any GW signal in its parameter space: $FF \geq M$
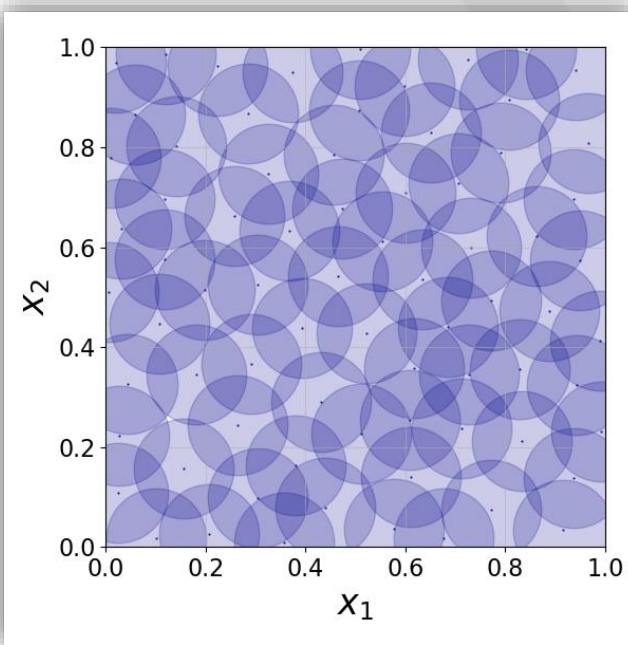 ➤ $M = 0.97$: minimal match
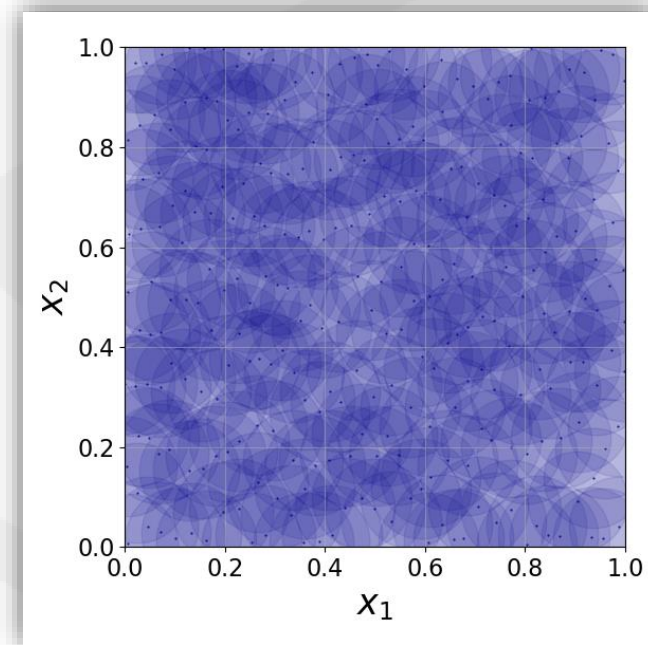☐ **Redundancy**
  ➤ More computational cost

☐ Take Euclidean 2D space as an example:

- **Inner product:**

$$(h \mid g) \equiv 4\Re \int_0^{+\infty} \frac{\tilde{g}^*(f)\tilde{h}(f)}{S_n(f)} \mathrm{d}f$$

- **Fitting Factor (FF):**

$$FF\left(\lambda^{\mu}\right) \equiv \max_{\lambda^{\mu}} \frac{\left(h(\lambda^{\mu}) \mid h(\lambda^{\mu'})\right)}{\sqrt{\left(h(\lambda^{\mu}) \mid h(\lambda^{\mu})\right)\left(h(\lambda^{\mu'}) \mid h(\lambda^{\mu'})\right)}}$$

Too few templates

"Just right"

Too many templates

Template Bank Generation

- A similar question with covering N-dimensional curved space *(signal manifold)*
- Hard to know optimal placement algorithms (exact metrics), especially higher dimension and curved space

- `sbank` package
  - Stochastic template bank placement

- **A(n over-)simplified version of `sbank`**
  - Only kept the key algorithms
  - Only consider Euclidean(-like) 2D space to make it simple and be illustrated easily
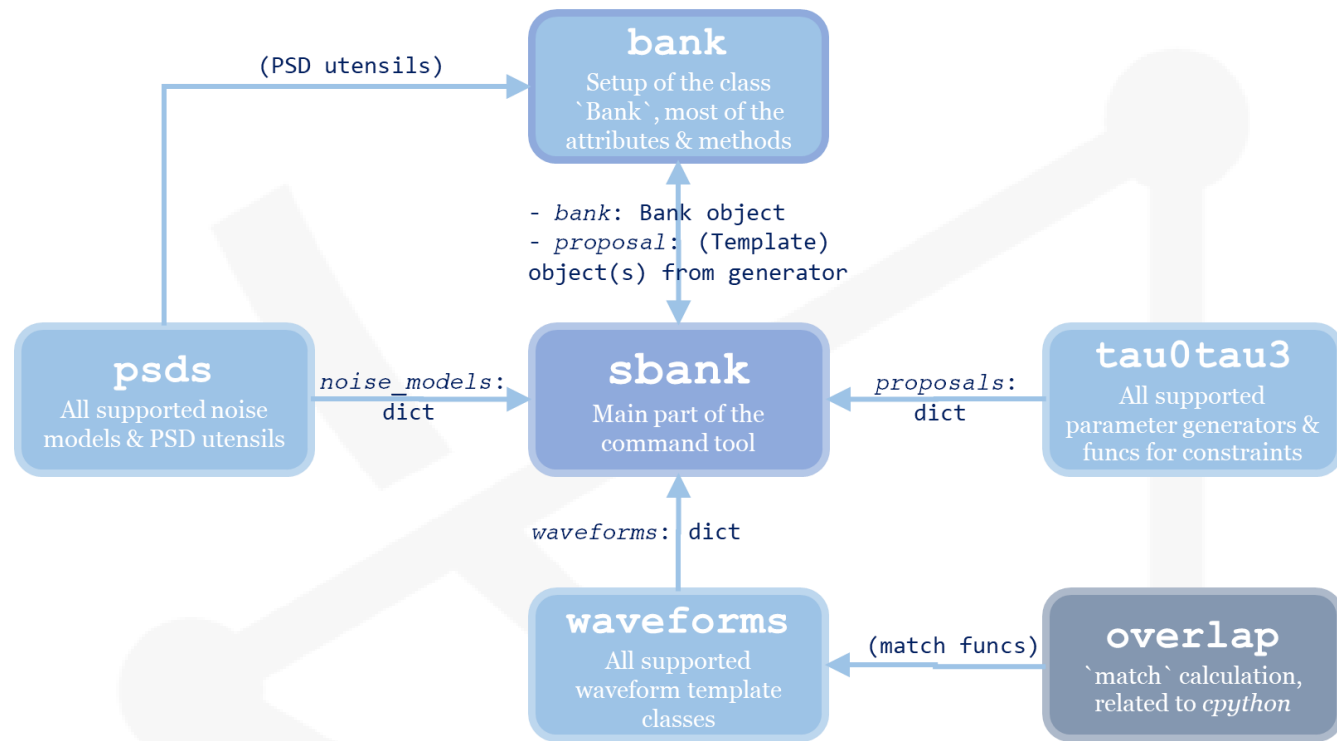
Structure of `sbank` package

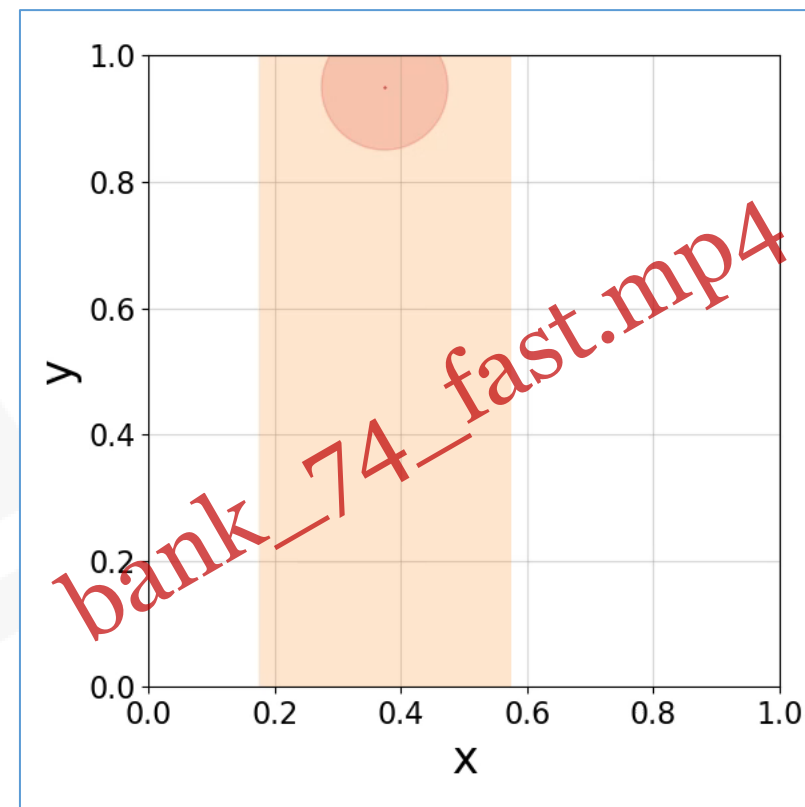[1] https://github.com/gwastro/sbank

Key Algorithm: Covering

☐ How should we regard one template as near the other one?

➢ Euclidean(-like) 2D space:
  ➢ Calculate the proper distance between two points
  ➢ Covering regions are drawn as circles(ellipses)
 ➢ (Here max_distance=0.1)

➢ Real GW template bank:
  ➢ Overlap:

$$O\left(h(\lambda^{\mu}), h(\lambda^{\mu'})\right) \equiv \frac{\left(h(\lambda^{\mu}) \mid h(\lambda^{\mu'})\right)}{\sqrt{\left(h(\lambda^{\mu}) \mid h(\lambda^{\mu})\right)\left(h(\lambda^{\mu'}) \mid h(\lambda^{\mu'})\right)}}$$

Key Algorithm: Covering

□ How can we filter proposals when generating a bank?
  ➢ We are not expected to calculate overlaps between a proposal and ALL templates in a bank
  ➢ Neighborhood parameter and neighborhood size
    ➢ Templates are sorted by this parameter

➢ Euclidean(-like) 2D space:
  ➢ (e. g.) Neighborhood parameter: the value of x
    ➢ (Here nhood_size=0.2)

➢ Real GW template bank:
  ➢ Neighborhood parameter:
    ➢ Tau0: A function of chirp mass and lower frequency
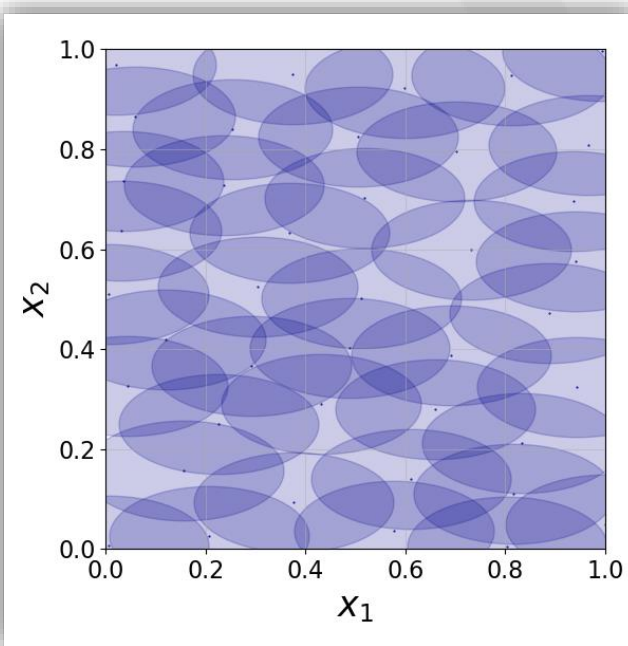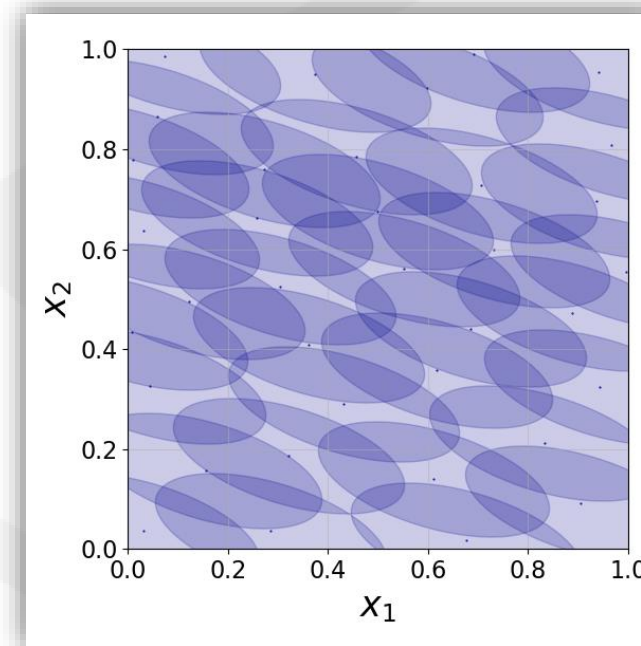   ➢ Since Chirp mass is a dominant parameter in GW
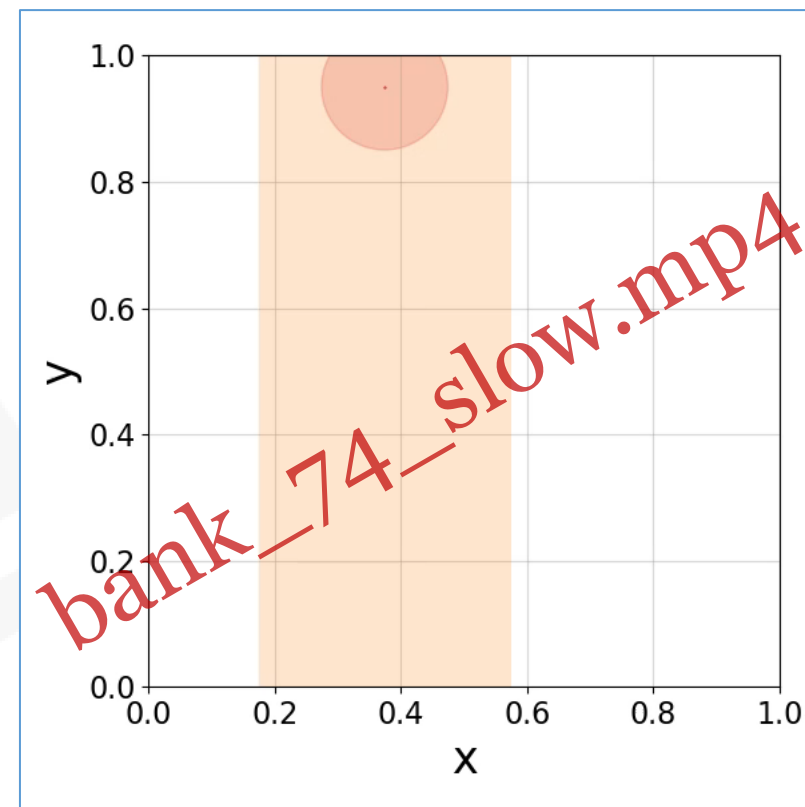
Further Discussions

- Does the distribution of trial points(proposals) effect template distribution?
  - Almost none. Better distribution can help converge faster

- Real GW template bank:
  - Need to consider the physical distribution when stochastically propose a parameter
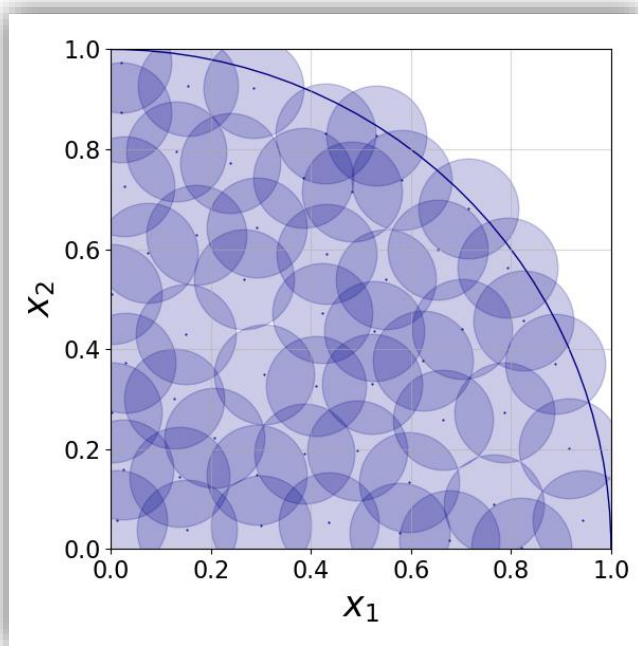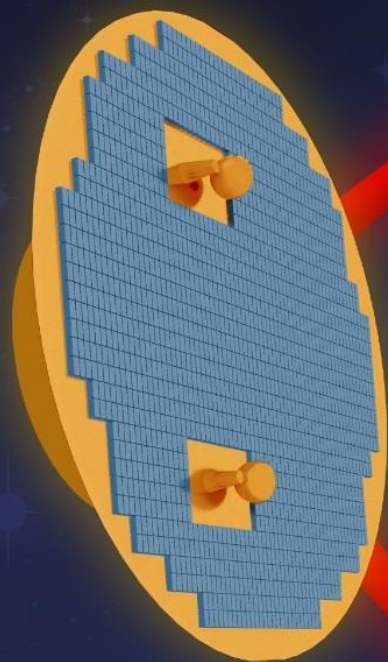

Cartesian metric but a polar generator
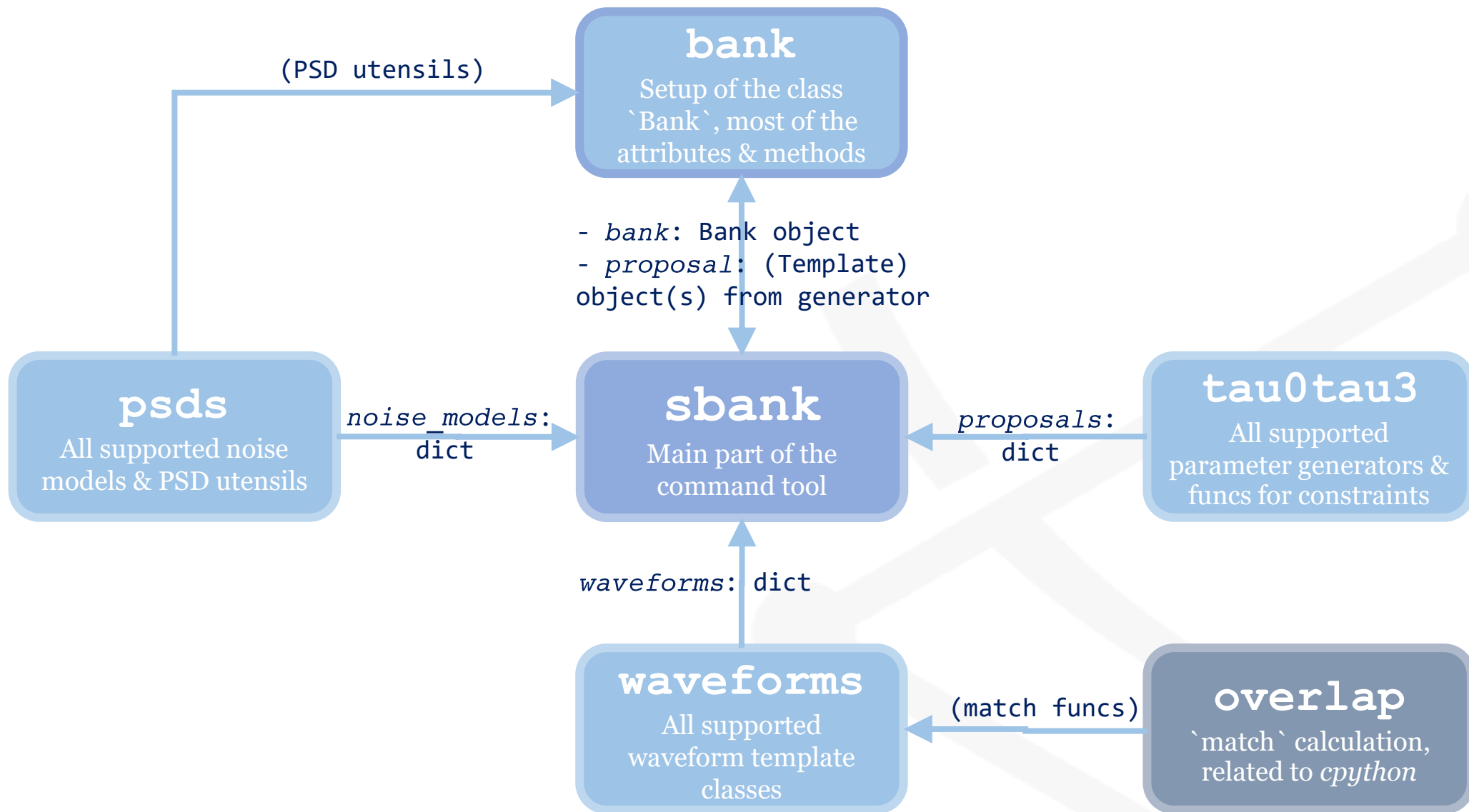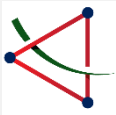
Metric: [[1/4, 0], [0, 1]]

Metric: [[1/4, 1/4], [1/4, 1]]

地球
Earth

Thanks

`sbank` structure

`sbank_simplified` structure

bank
Setup of the class
`Bank`, most of the
attributes & methods

- *bank*: Bank object
- *proposal*: (Template)
object(s) from generator

Matched_filtering
Additional codes for an illustration

sbank
Main part of the
command tool

proposals: dict
coord_frames: dict

templates
All supported
template classes

generators:
All supported 2D
point generators

proper_distance:
$dl = \sqrt{g_{ij} dx^i dx^j}$