Implementing parameter estimation for CBC sources using particle swarm optimization

Hyung Won Lee, Inje University
(collaboration with Jeongcho Kim, NIMS)
January 29, 2025, ICRR

Approved budget

- No budget was requested for year 2024

Research purpose

- LIGO/Virgo/KAGRA collaboration are detecting gravitational wave events, especially from compact binary coalescence (CBC) sources.
- The number of events will be increased with improved sensitivities and it is necessary to develop more rapid parameter estimation to obtain useful astrophysical results timely.
- Developed parameter estimation methods based on MCMC or nested sampling, which is relatively slow to get the final results.

Research purpose

- I want to develop a parameter estimation method using particle swarm optimization (PSO) methods and implement to KAGALI (KAGRA Algorithmic Library) for parameter estimation pipeline.
- Success of KAGRA detector is very important and give milestone for next generation gravitational wave detectors.
- KAGRA's unique character like underground and cryogenic will supply many knowhows to future detector design and managements.

2024 Plan

- In year 2024, I will develop C-code implementation of particle swarm optimization on KAGALI environment or lalsuite environment depending on the KAGRA strategy.
- This implementation will be tested by some injection studies to decide its usability for parameter estimation.
- If this is developed successfully, KAGRA will have its own parameter estimation pipeline independently, which could be substantial contribution to gravitational wave science.

Contents

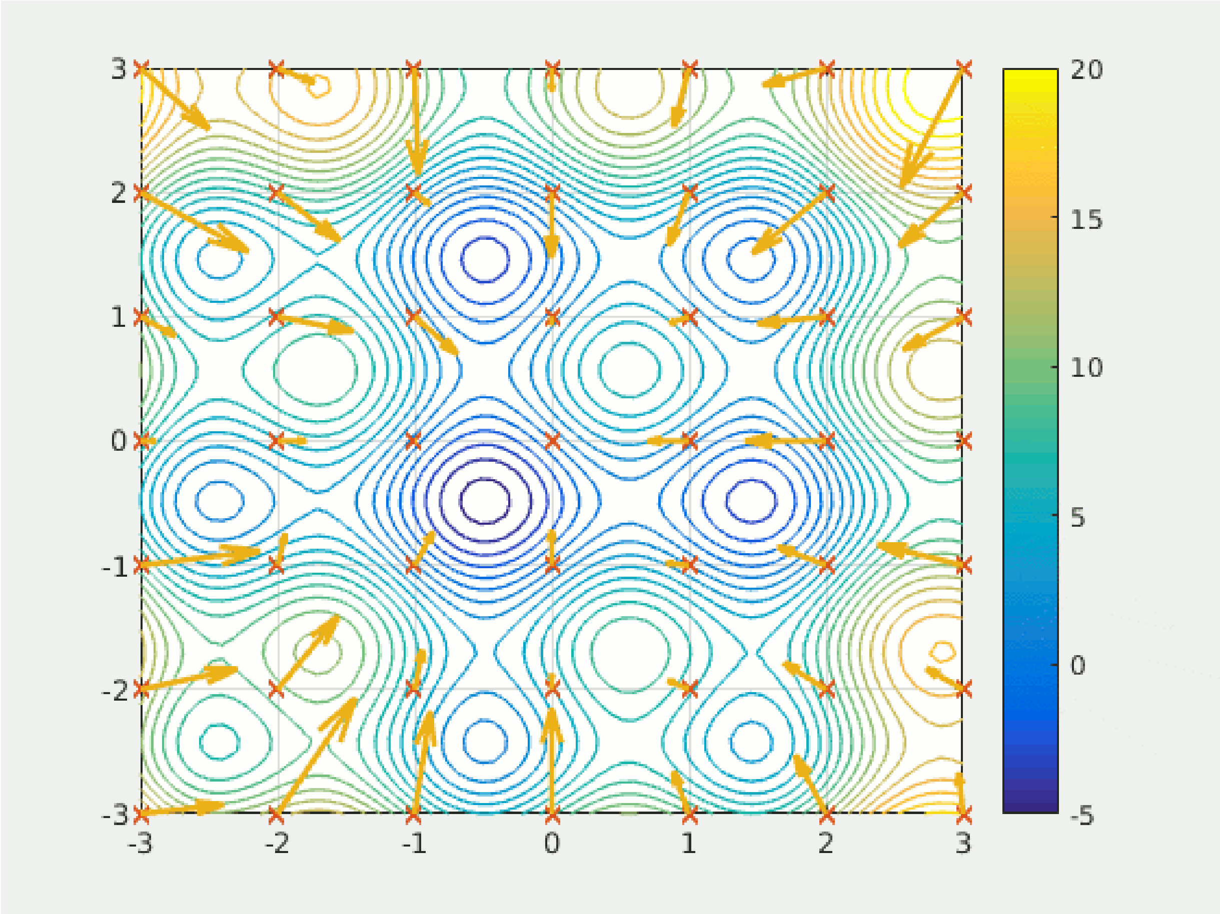
- Motivation
- PSO method brief
- What's new
- Implementations
- Future plans

Motivation

- Long evaluation time for full posterior calculation
- Can we shorten PE time?
- PSO looks global optimal value
- Find the global optimal value, then evaluate posterior around that point with limited parameter space

PSO method brief

- computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality



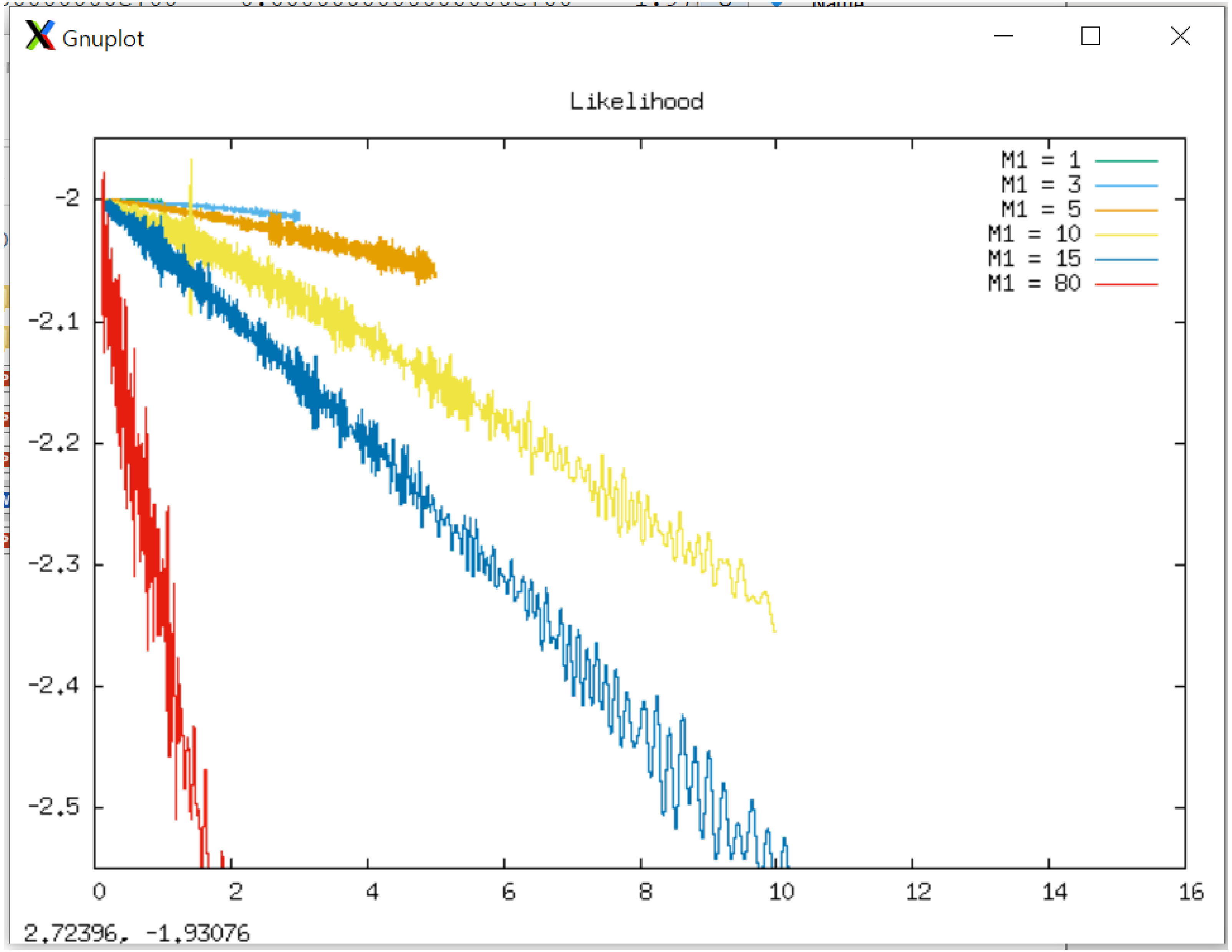
Credit: Wikipedia

Basic Algorithm

```
for each particle i = 1, ..., S do
    Initialize the particle's position with a uniformly distributed random vector: xi ~ U(blo, bup)
    Initialize the particle's best known position to its initial position: pi ← xi
    if f(pi) < f(g) then
        update the swarm's best known position: g ← pi
    Initialize the particle's velocity: vi ~ U(-|bup-blo|, |bup-blo|)
while a termination criterion is not met do:
    for each particle i = 1, ..., S do
        for each dimension d = 1, ..., n do
            Pick random numbers: rp, rg ~ U(0,1)
            Update the particle's velocity: vi,d ← w vi,d + φp rp (pi,d-xi,d) + φg rg (gd-xi,d)
        Update the particle's position: xi ← xi + vi
        if f(xi) < f(pi) then
            Update the particle's best known position: pi ← xi
            if f(pi) < f(g) then
                Update the swarm's best known position: g ← pi
```

$$v_d = c_1 v_d + c_{max} r_1 (p_d - x_d) + c_{max} r_2 (g_d - x_d)$$
$$x_d = x_d + v_d$$

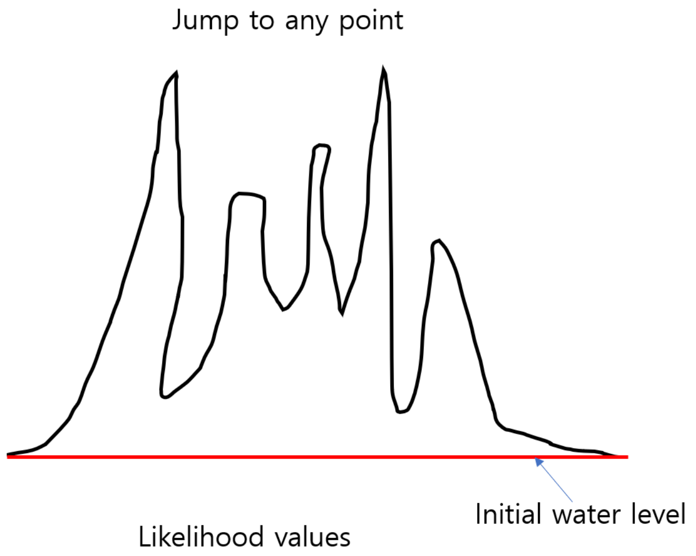
Likelihood not Smooth

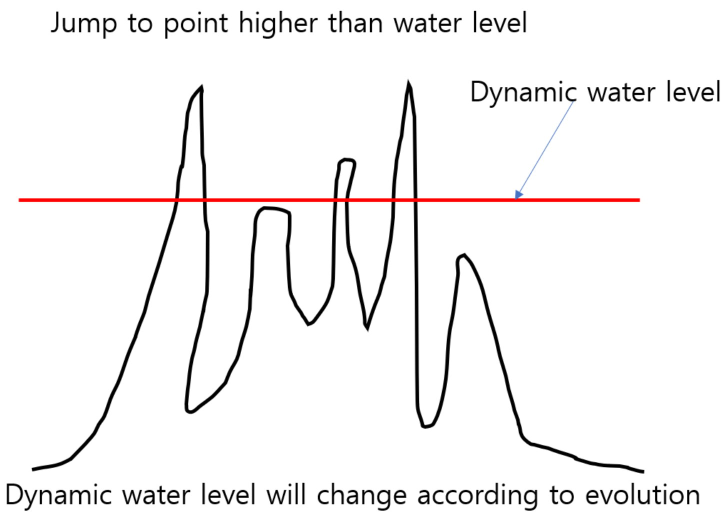


What's new

- Usual Likelihood values are very spiky
- Difficult to find the global maximum

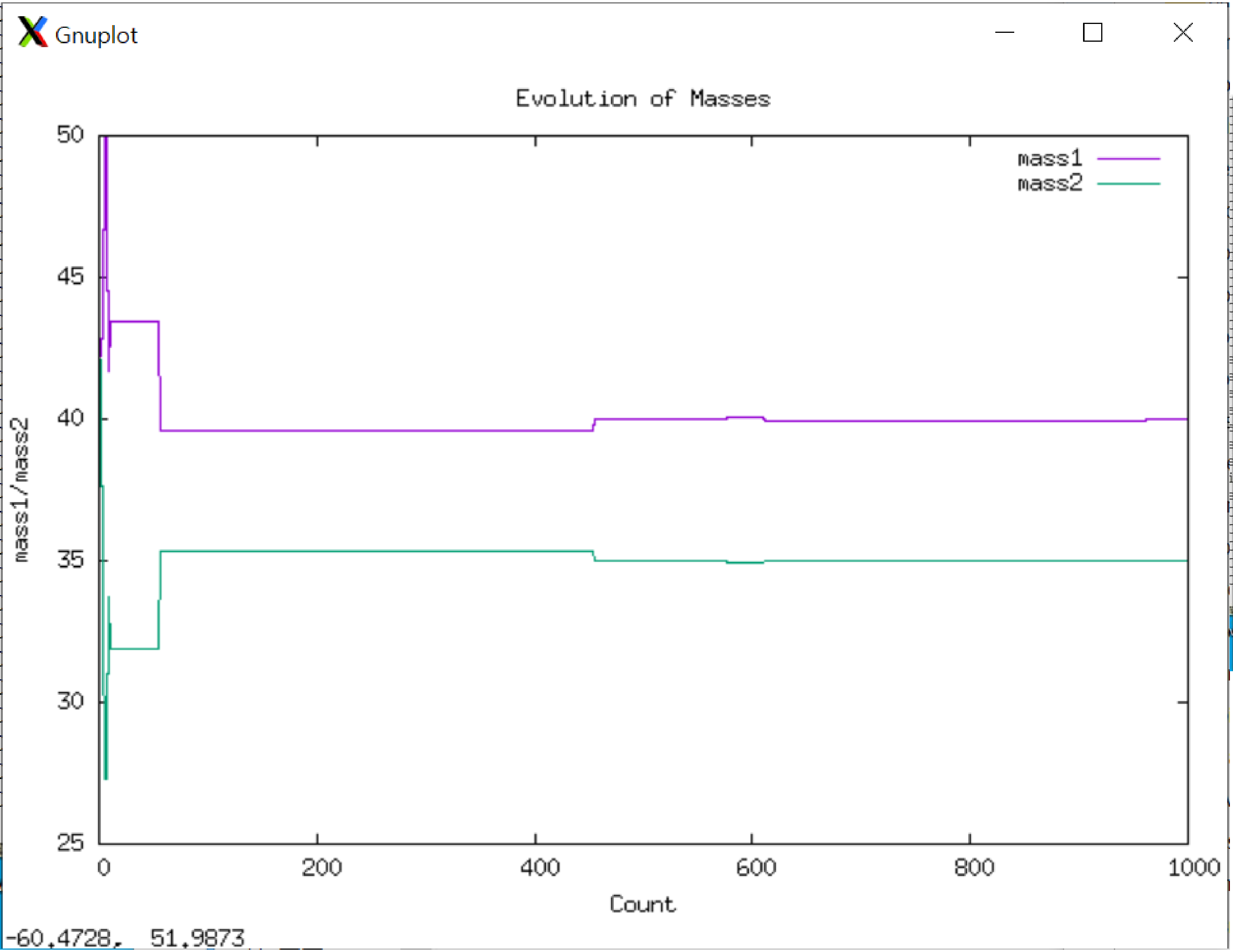
New PSO algorithm

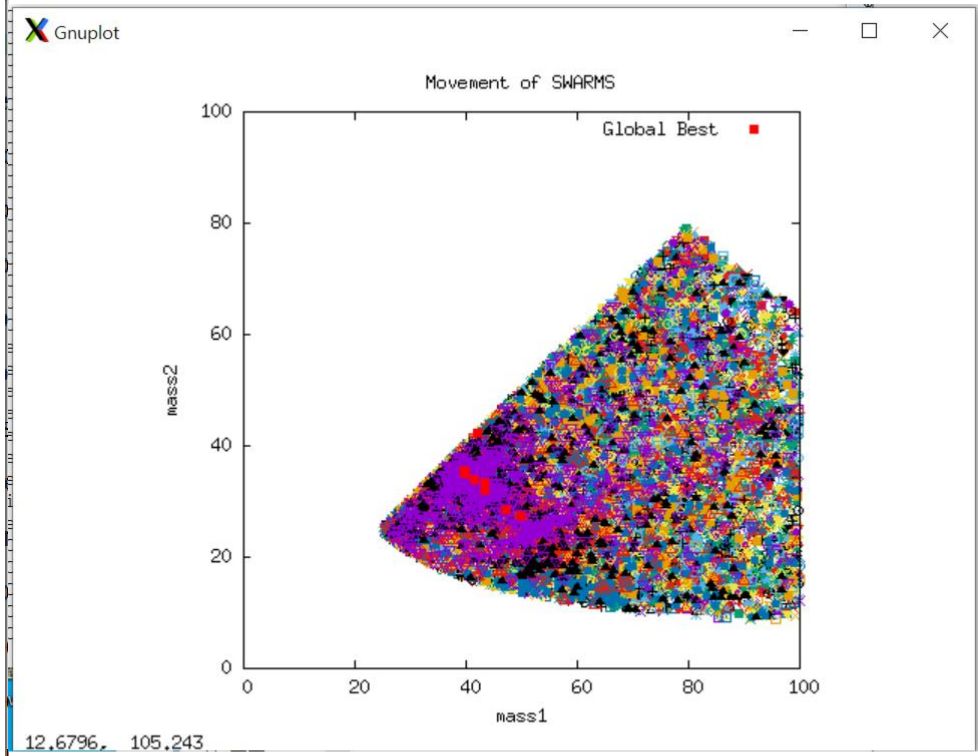


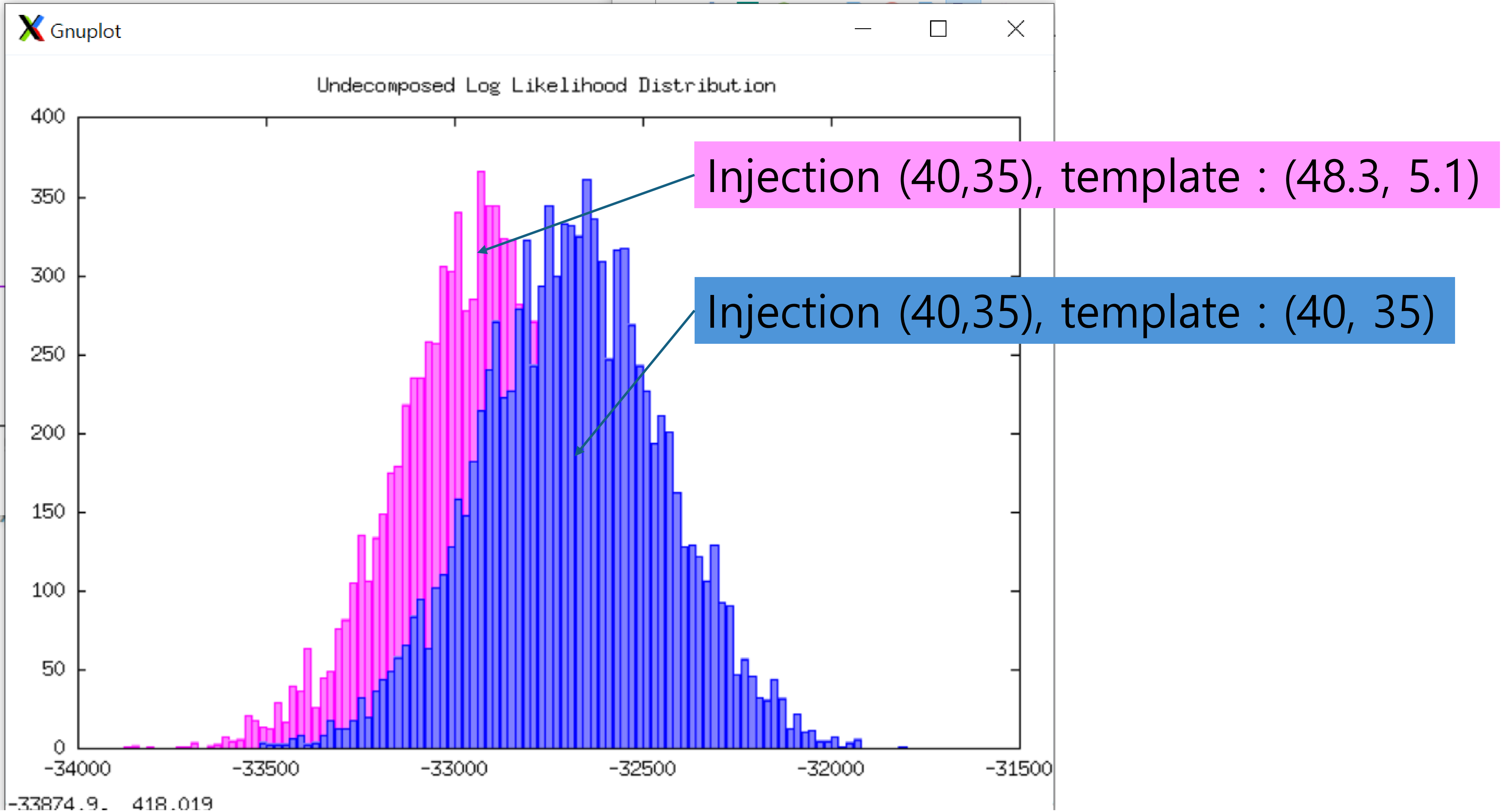


Implementations

- Working on KISTI local branch based on lalsuite
- Branches
  - gw_pso
  - normalized_likelihood


Gnuplot
Evolution of Masses
mass1
mass2
mass1/mass2
Count
-60.4728, 51.9873


Gnuplot
Movement of SWARMS
Global Best
mass2
mass1
12.6796, 105.243


Gnuplot
Undecomposed Log Likelihood Distribution
400
350
300
250
200
150
100
50
0
-34000
-33500
-33000
-32500
-32000
-31500
-33874.9, 418.019
Injection (40,35), template : (48.3, 5.1)
Injection (40,35), template : (40, 35)

Future Plans

- Find robust measure function
- Injection study

Thanks