

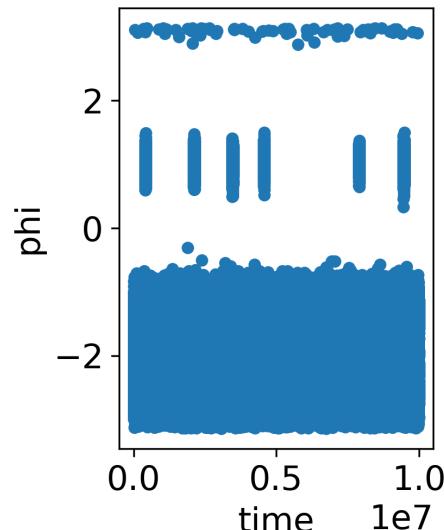
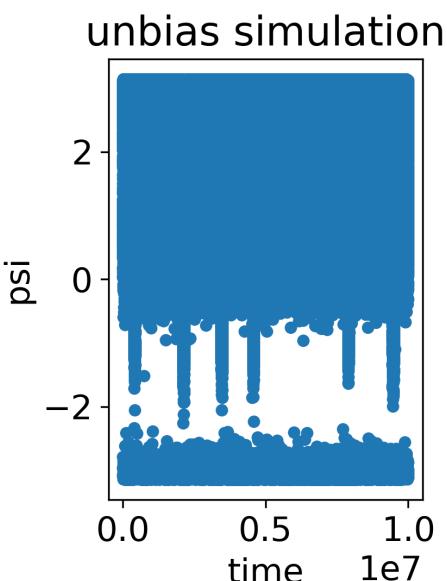
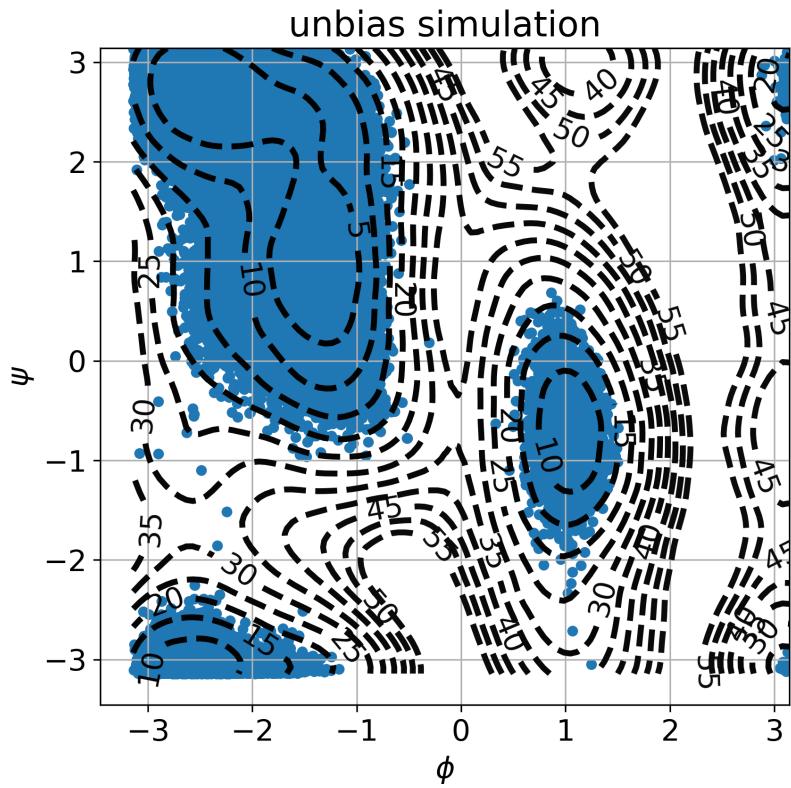
Different tprime evaluation on Biased trajectories of Alanine Dipeptide

The main goal of this analysis is to show that different rescaled time will lead to different estimates of the time matrix correlation.

In particular it is clear that working with a not rescaled time is equal to work with the rescaled one when few transitions between metastable states occur.

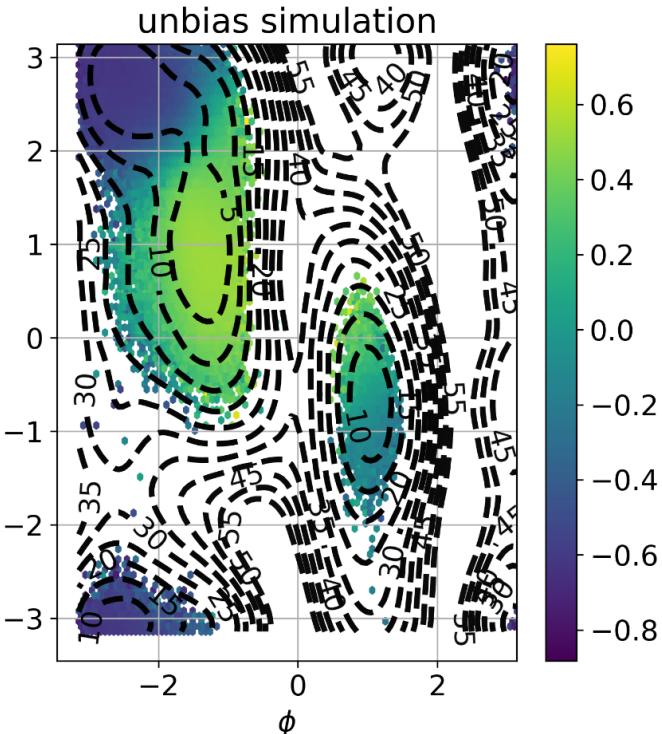
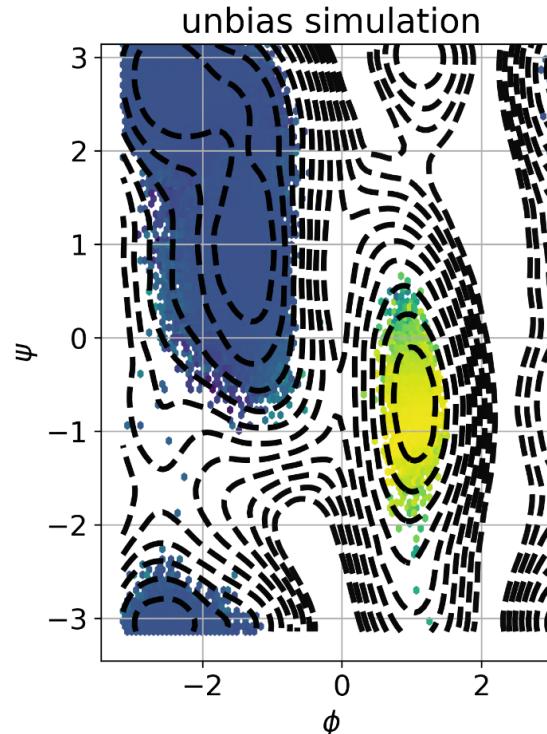
On the other hand when the applied bias forces the system to explore the phase space in a diffusive way, and many transitions between metastable states are observed, The Deep-TICA Analysis performed with the rescaled time is not able to identify the slow modes of the system. In contrast the use of a not rescaled time (and a much weaker rescaled time) can at least detect the slowest mode of the system under study (transition between state A and state B of Alanine).

Long Unbiased trajectory 10 mus

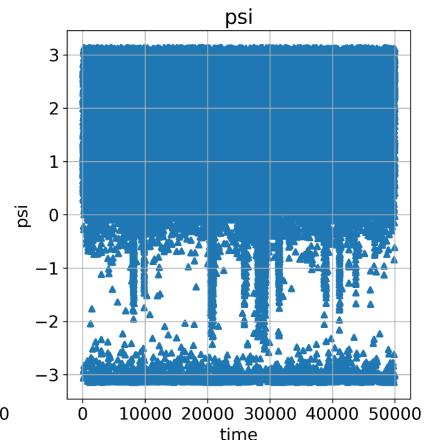
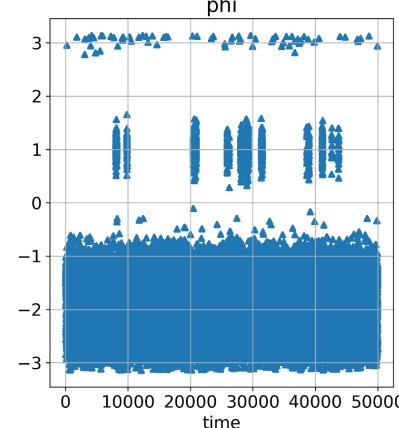
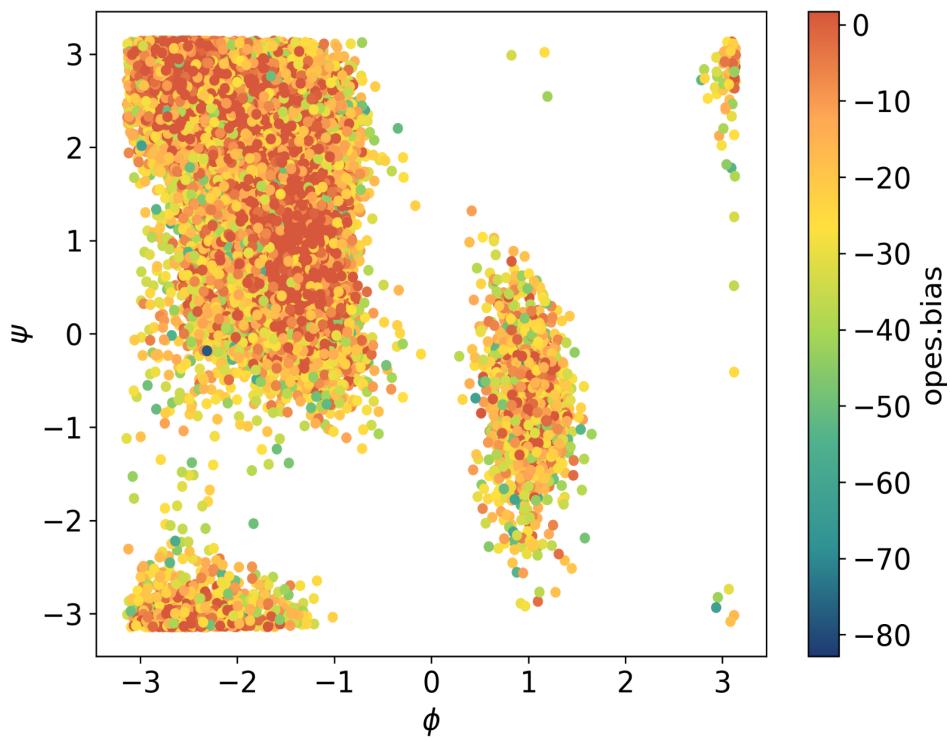


Deep TICA Analysis

lag time = 20ps (stride of simulation). With this stride is not possible to detect properly the second slowest mode



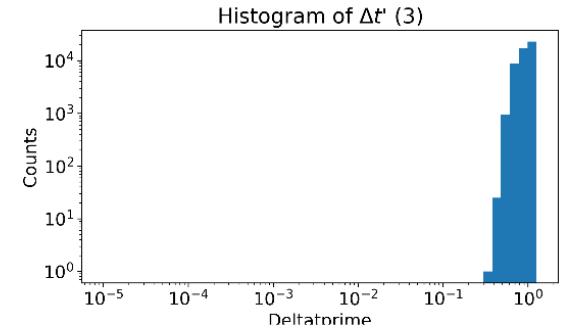
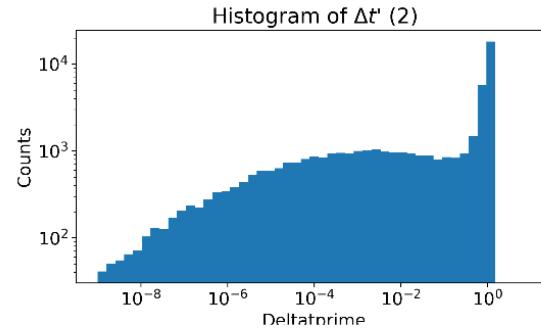
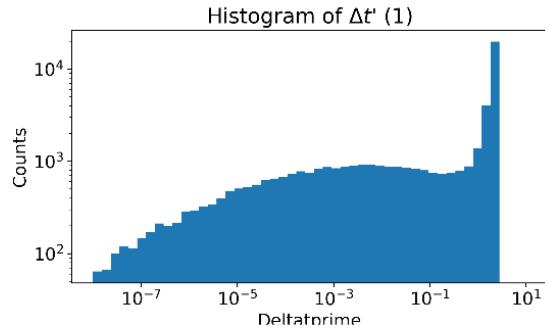
Biased trajectory, multi Thermal 50ns



Tprime and logweights, different set ups

```
1 logweights = beta*bias  
1 tprime = dt * cumsum(exp(logweights))  
  
2 logweights = beta*bias - max(beta*bias)  
2 tprime = dt * cumsum(exp(logweights))  
  
3 logweights = beta*bias - max(beta*bias)  
3 logweights /= abs(min(logweights))  
3 tprime = dt * cumsum(exp(logweights))
```

Distribution of Delta tprime

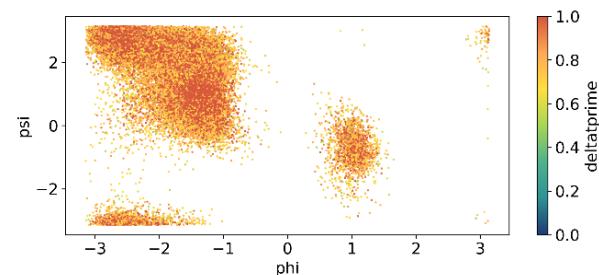
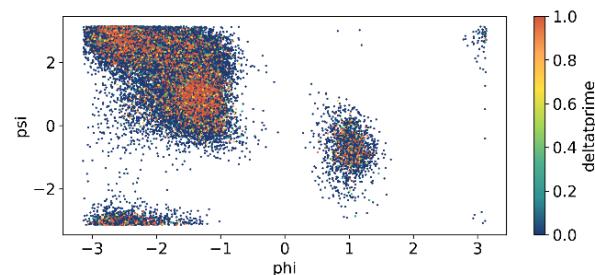
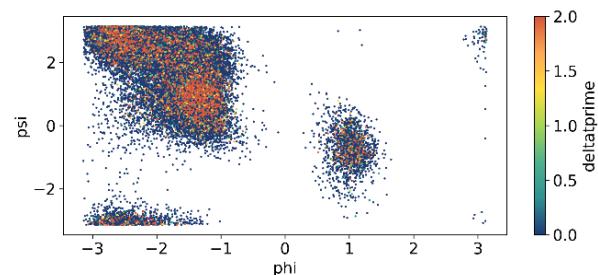


Lag time = 2

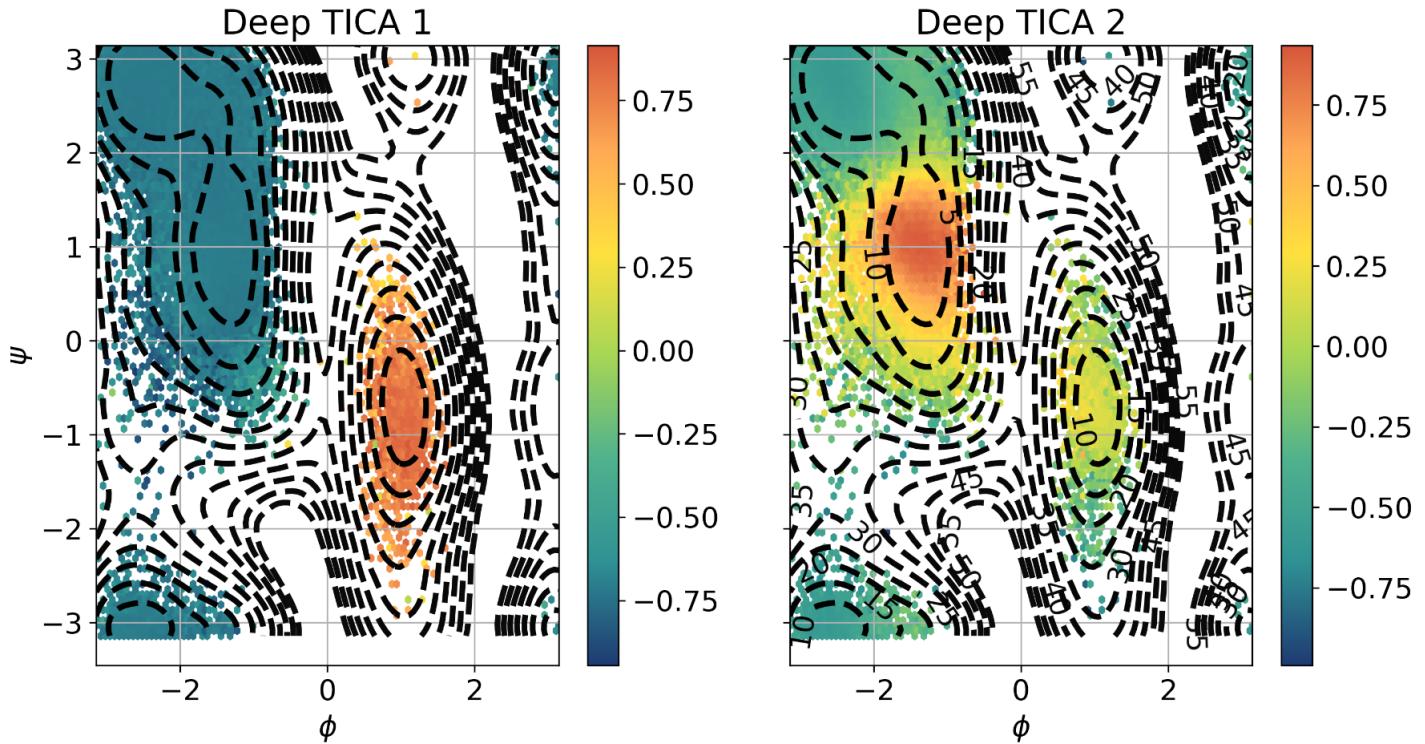
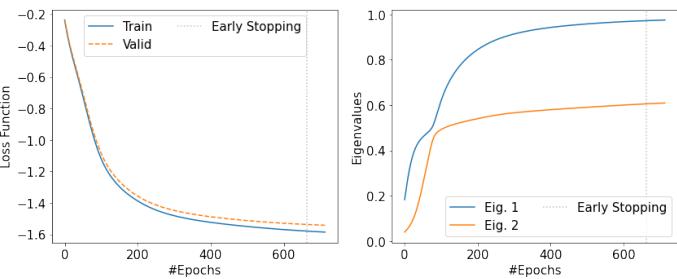
Lag time = 1

Lag time = 1

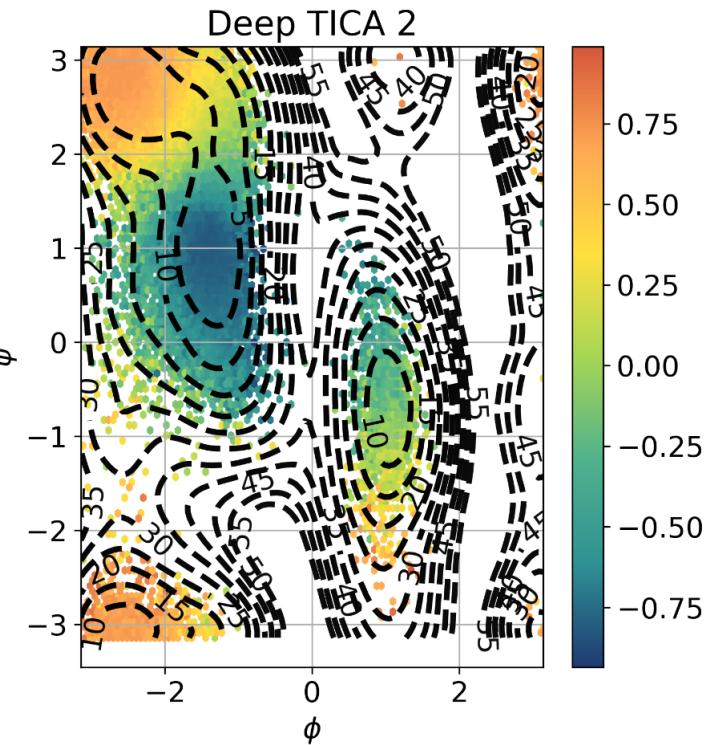
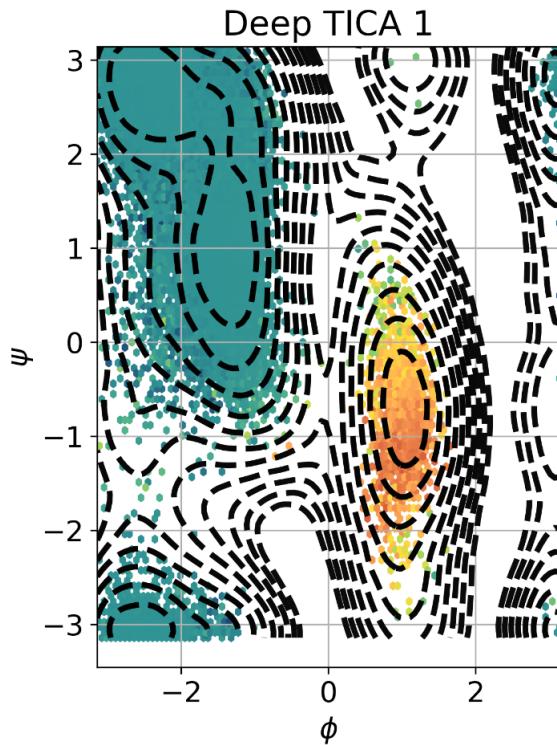
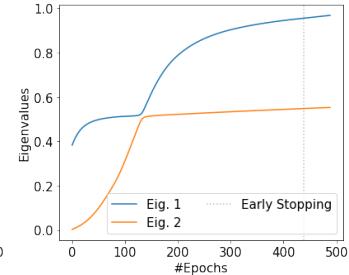
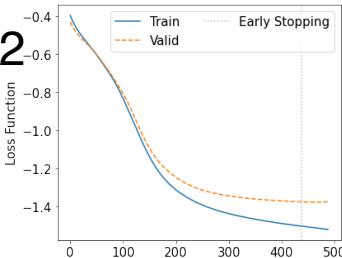
Trajectories



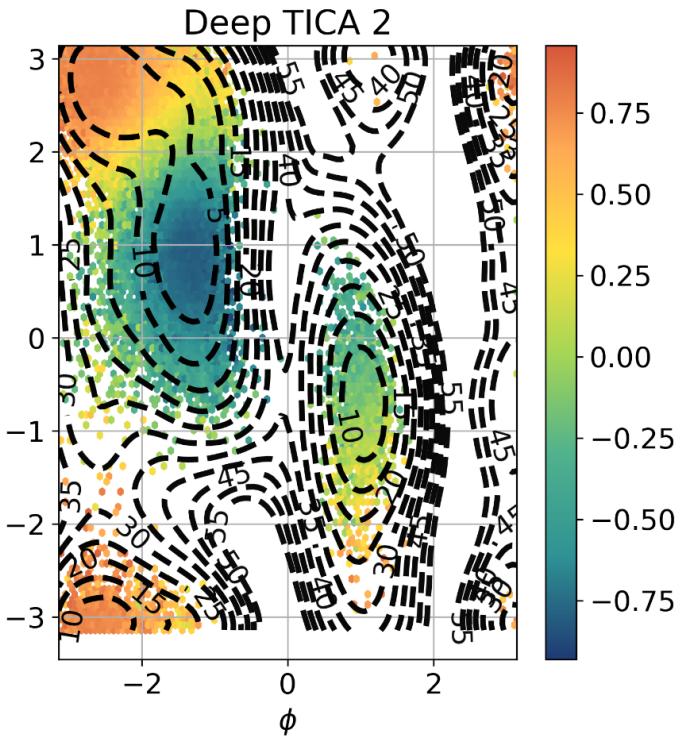
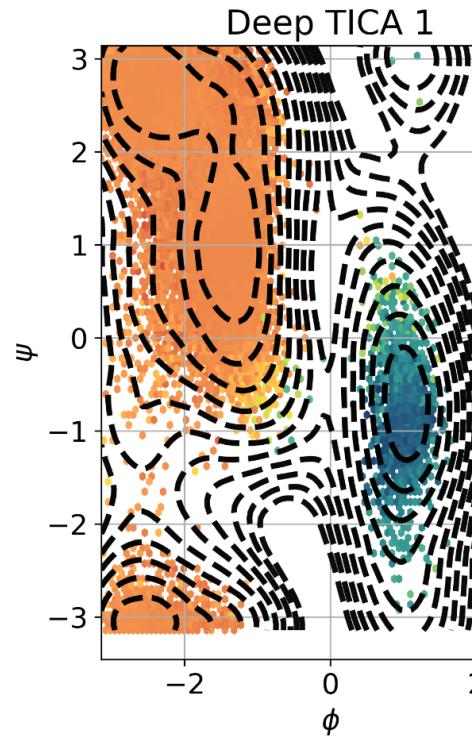
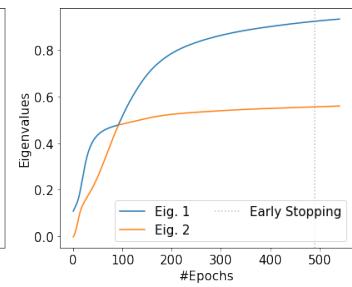
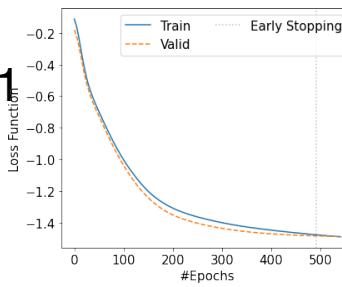
Deep TICA Analysis, without using the rescaled time, lag time = 1



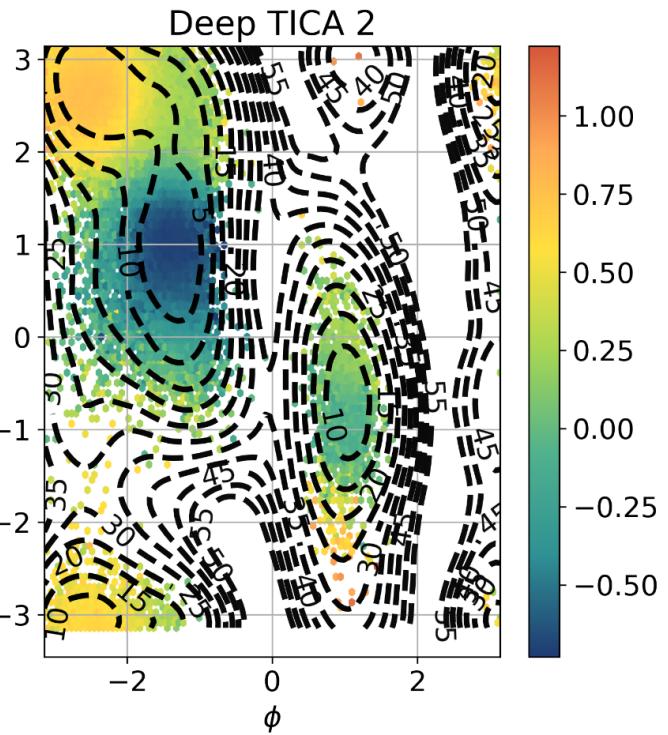
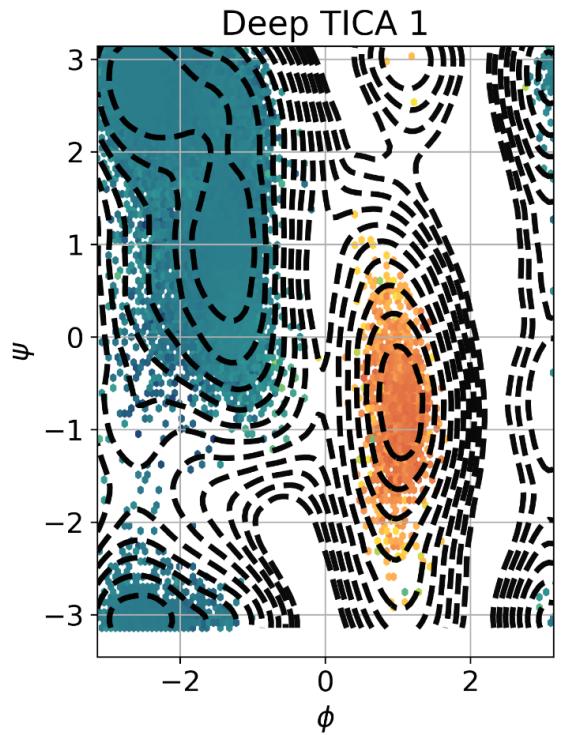
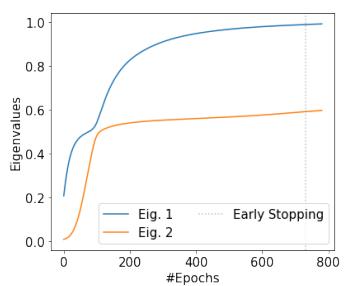
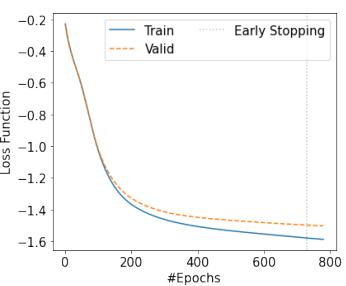
Deep TICA Analysis, rescaled as option (1) ,lag time = 2



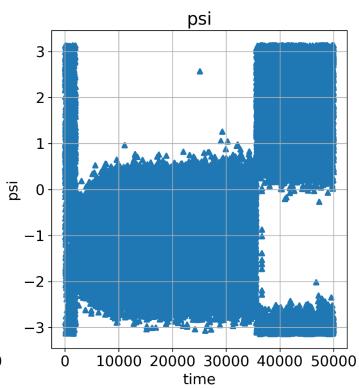
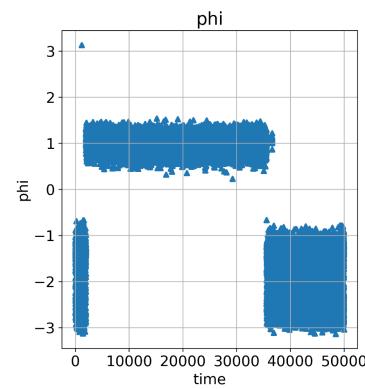
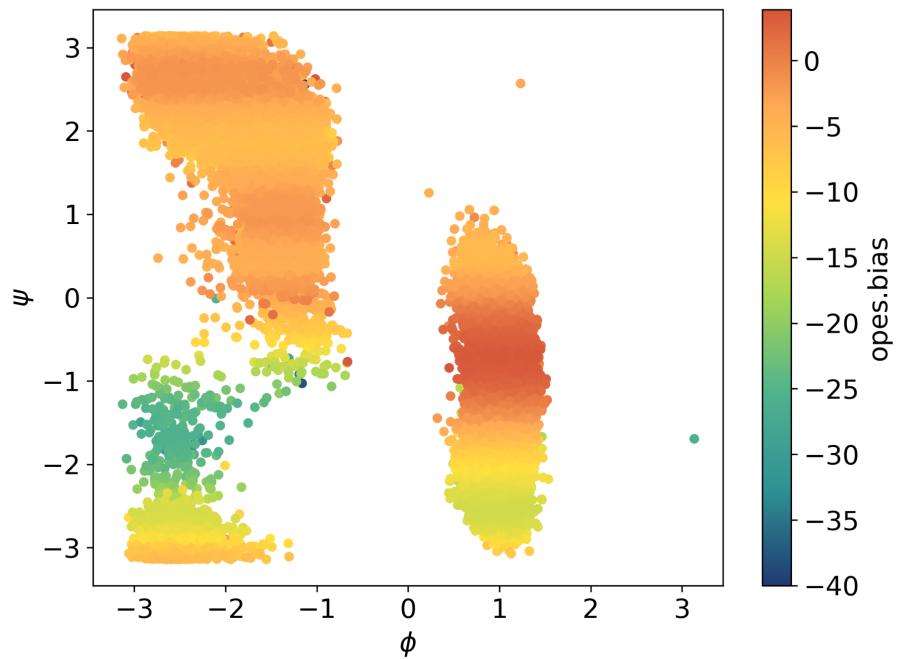
Deep TICA Analysis, rescaled as option (2) ,lag time = 1



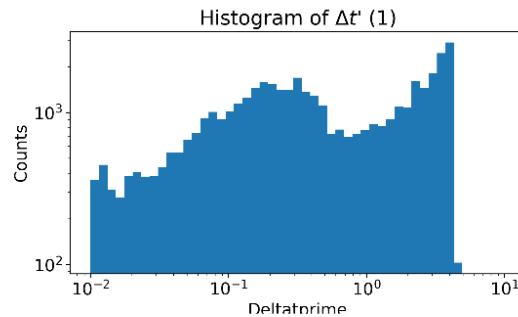
Deep TICA Analysis, rescaled as option (3) ,lag time = 1



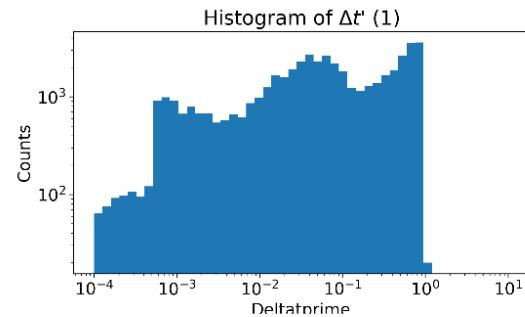
Biased trajectory, bias along Psi 50ns



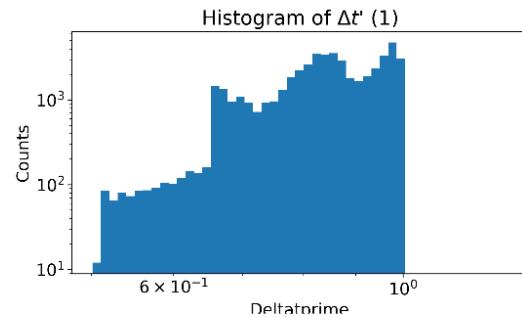
Distribution of Delta tprime



Lag time = 4

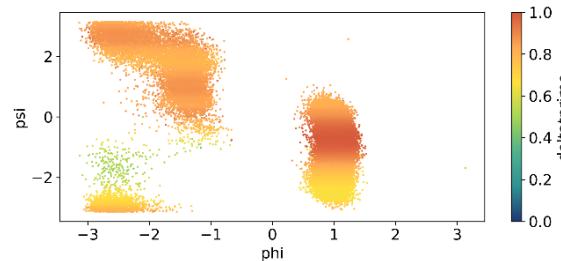
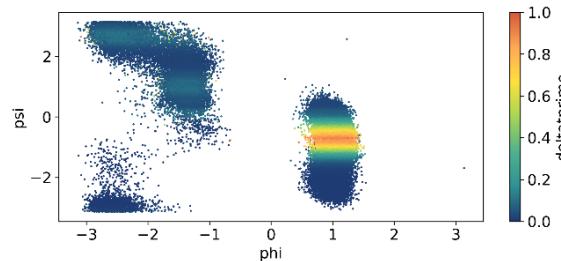
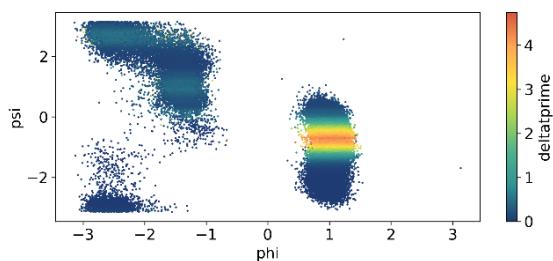


Lag time = 1

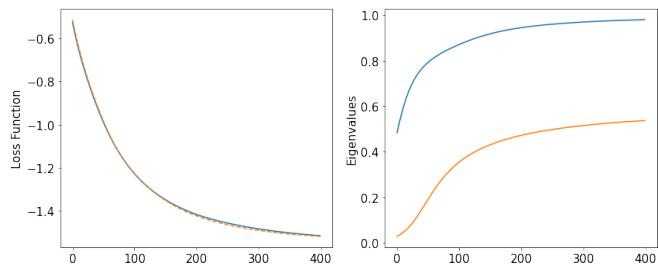
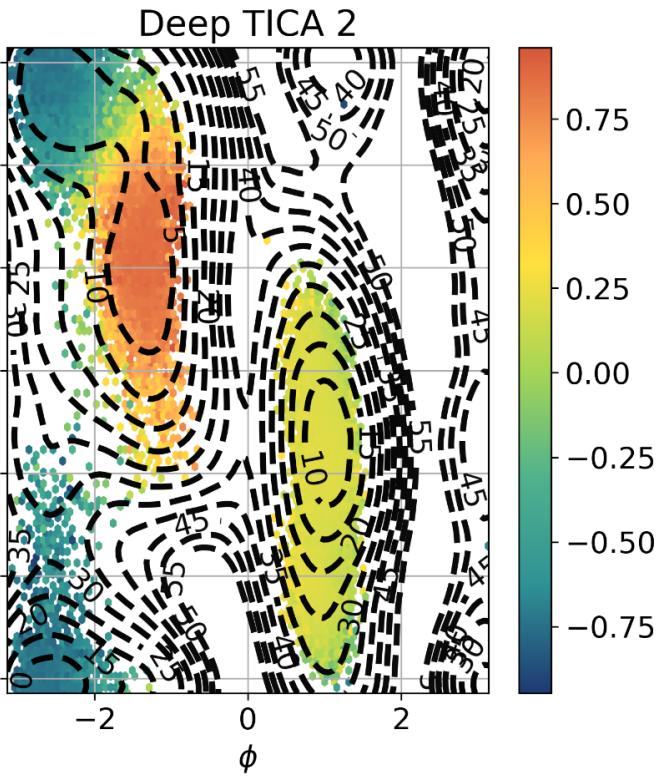
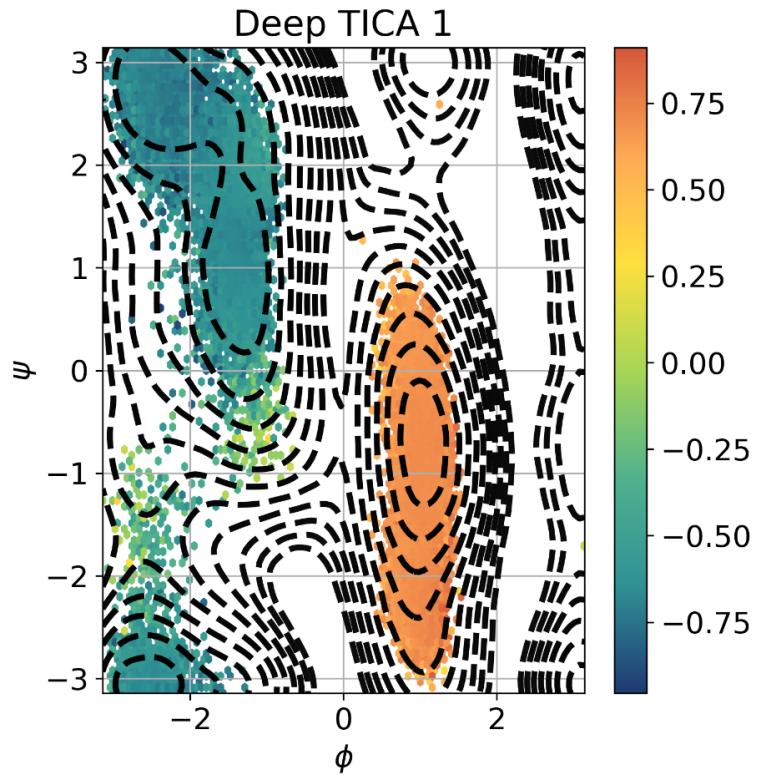


Lag time = 1

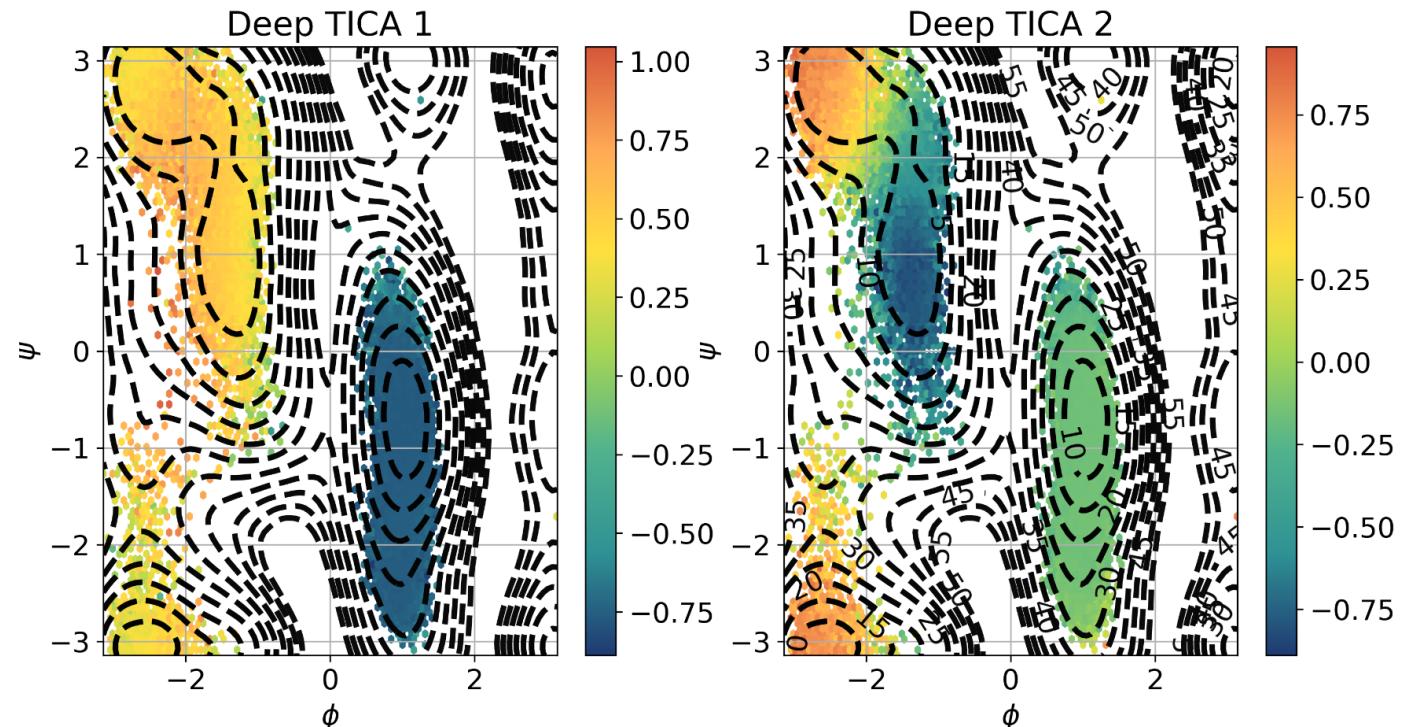
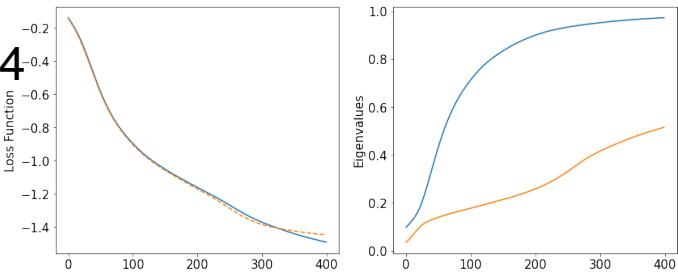
Trajectories



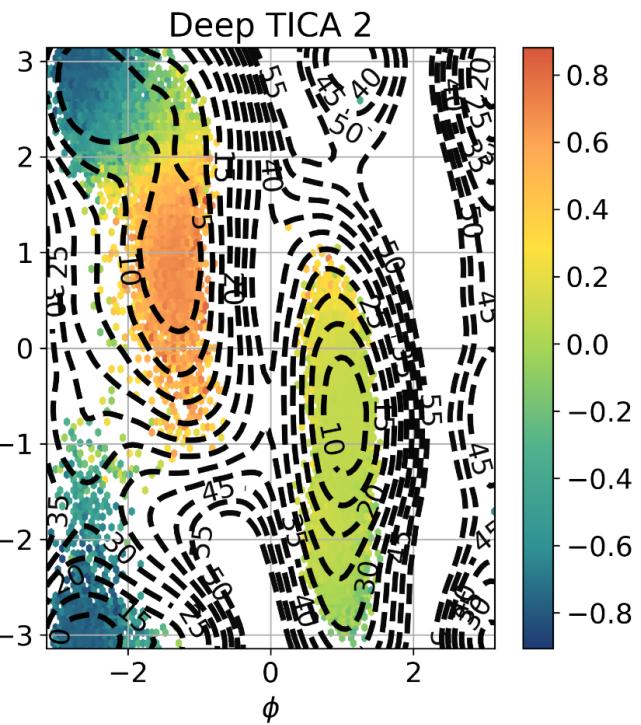
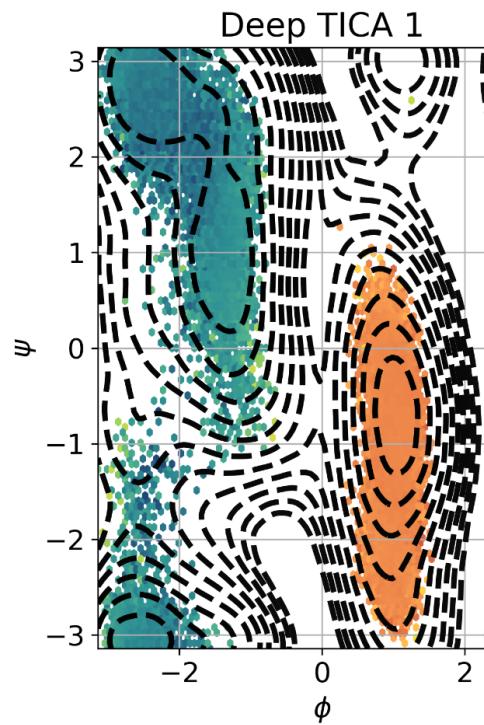
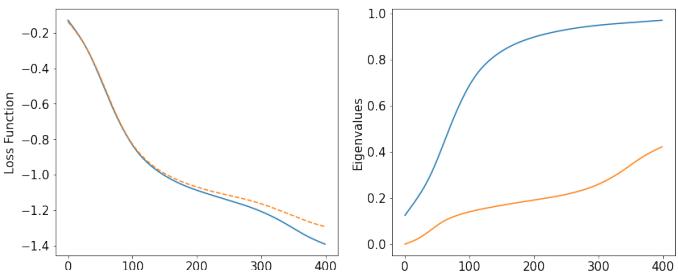
Deep TICA Analysis, without using the rescaled time, lag time = 1



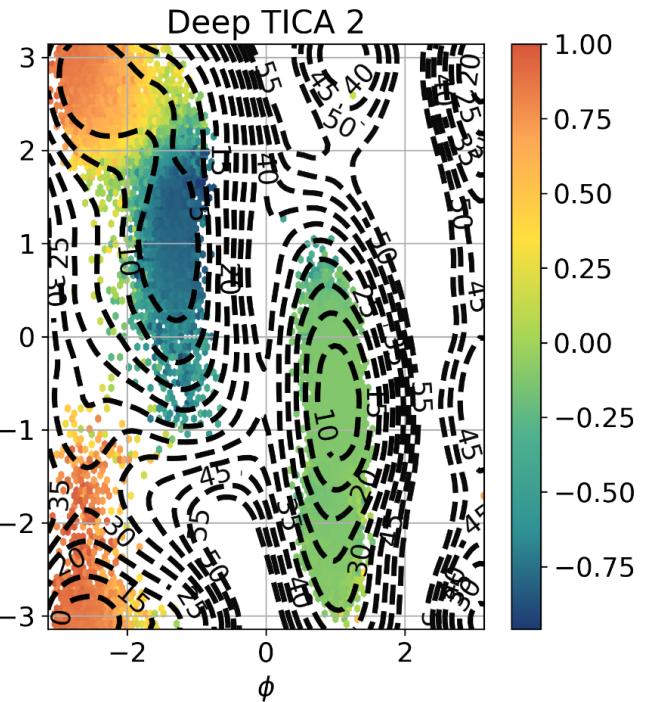
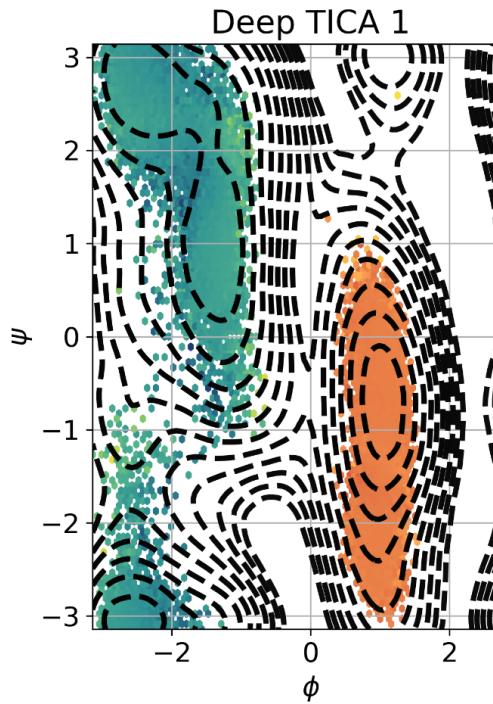
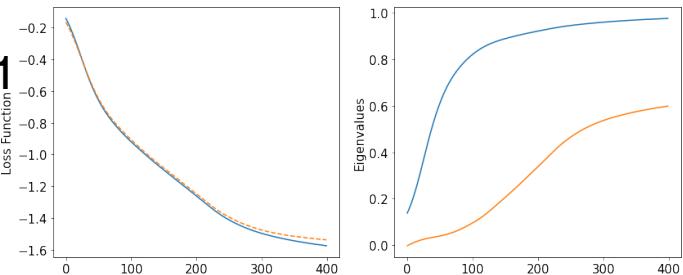
Deep TICA Analysis, rescaled as option (1) ,lag time = 4



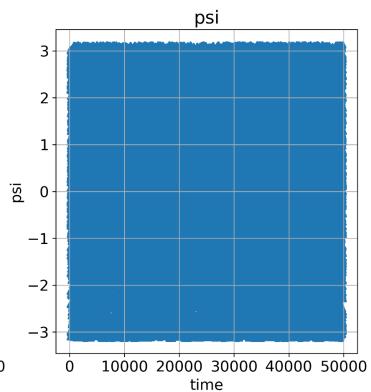
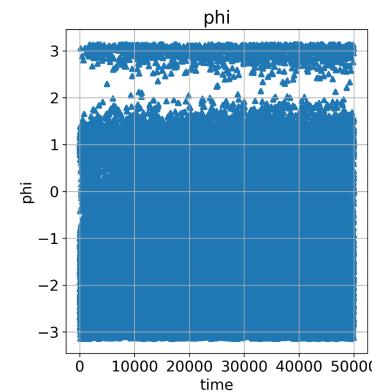
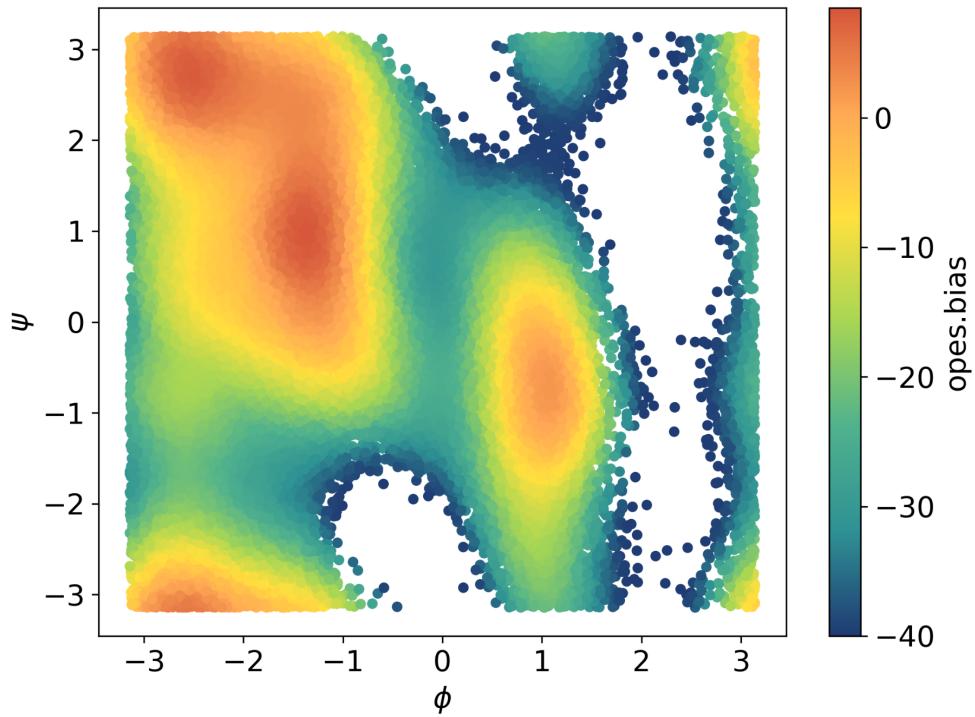
Deep TICA Analysis, rescaled as option (2) ,lag time = 1



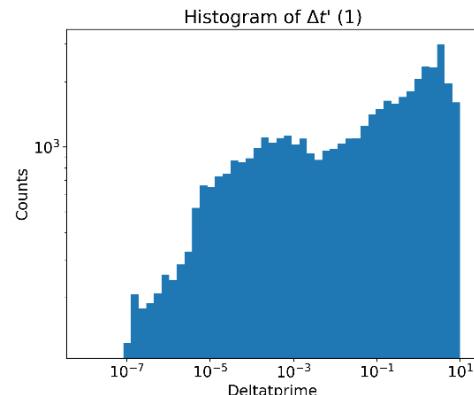
Deep TICA Analysis, rescaled as option (3) ,lag time = 1



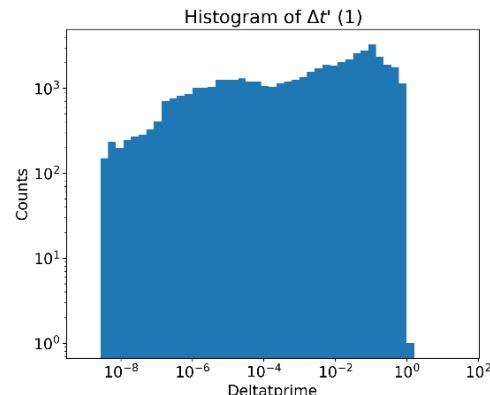
Biased trajectory, bias along both Psi and phi, 50ns, diffusive dynamics



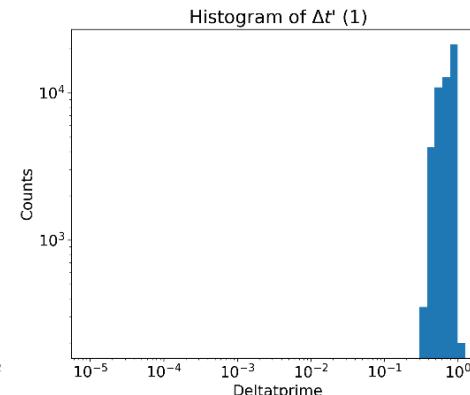
Distribution of Delta tprime



Lag time = 25

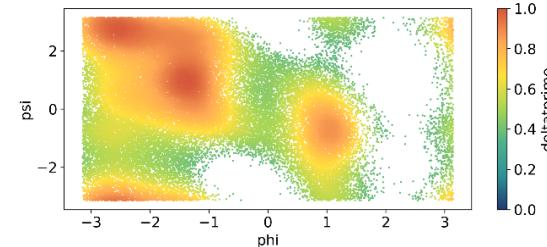
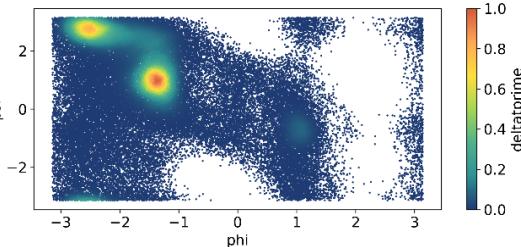
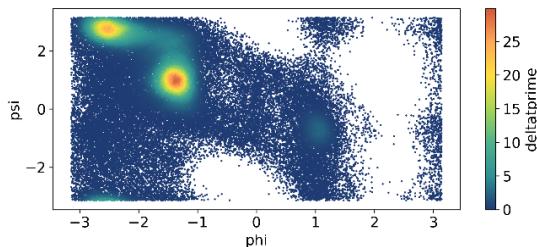


Lag time = 1

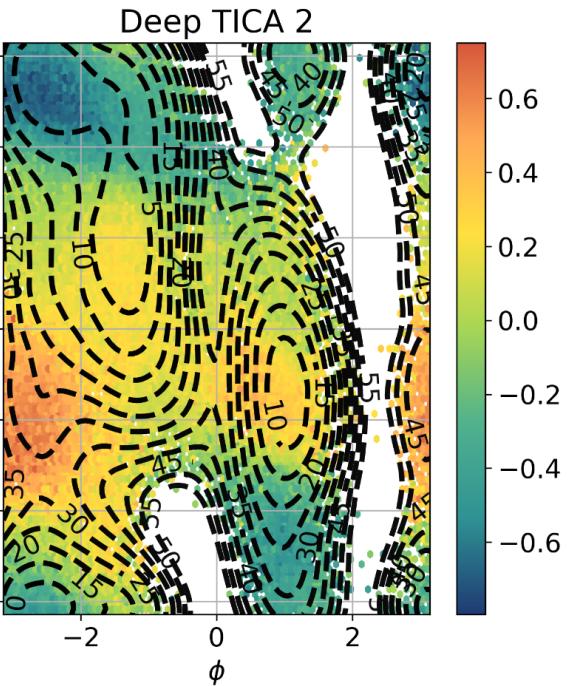
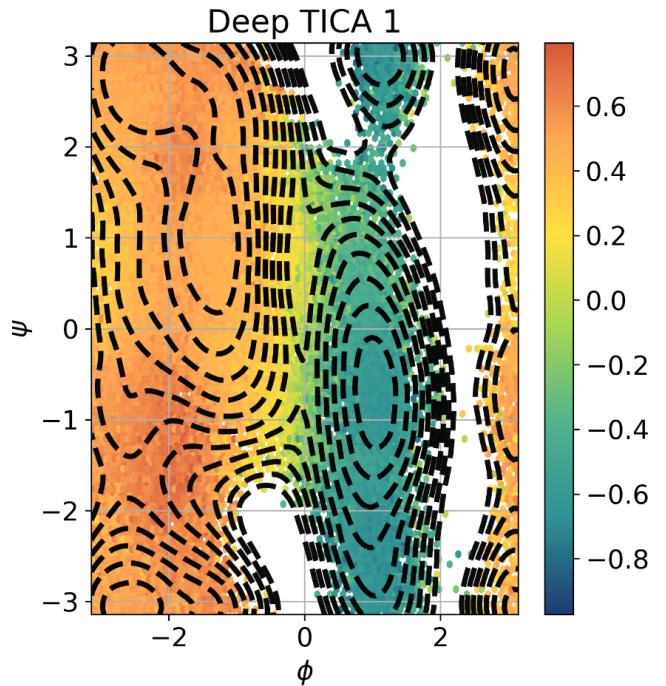
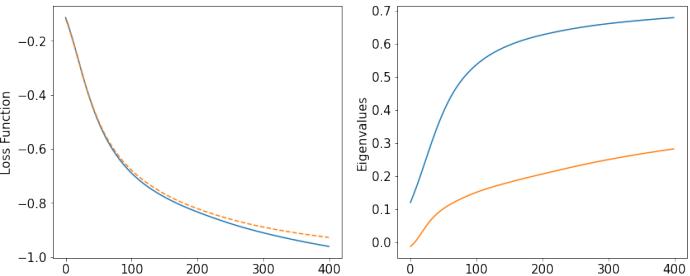


Lag time = 1

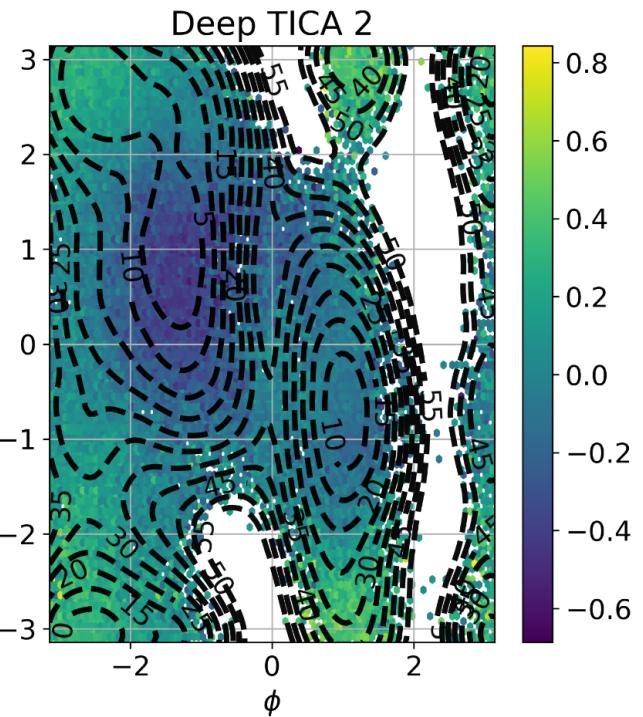
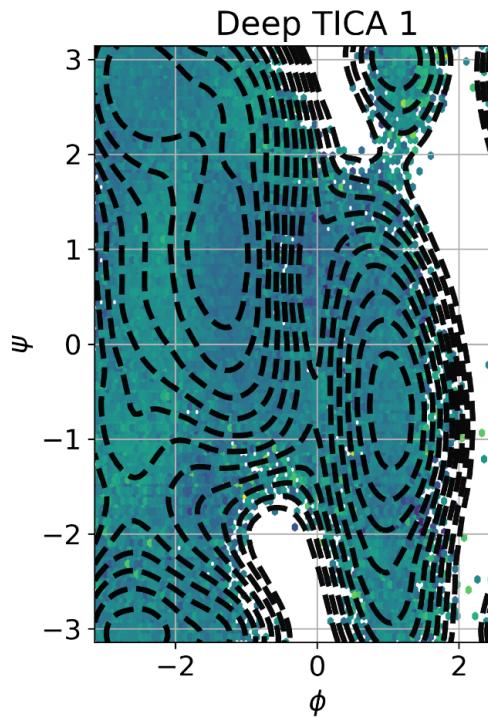
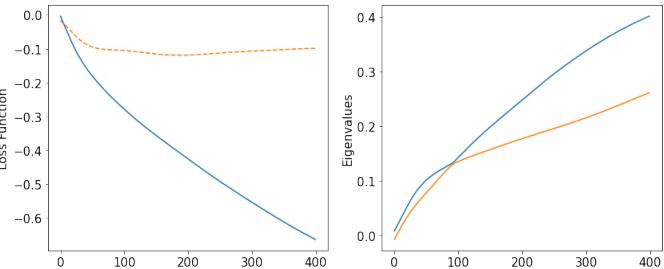
Trajectories



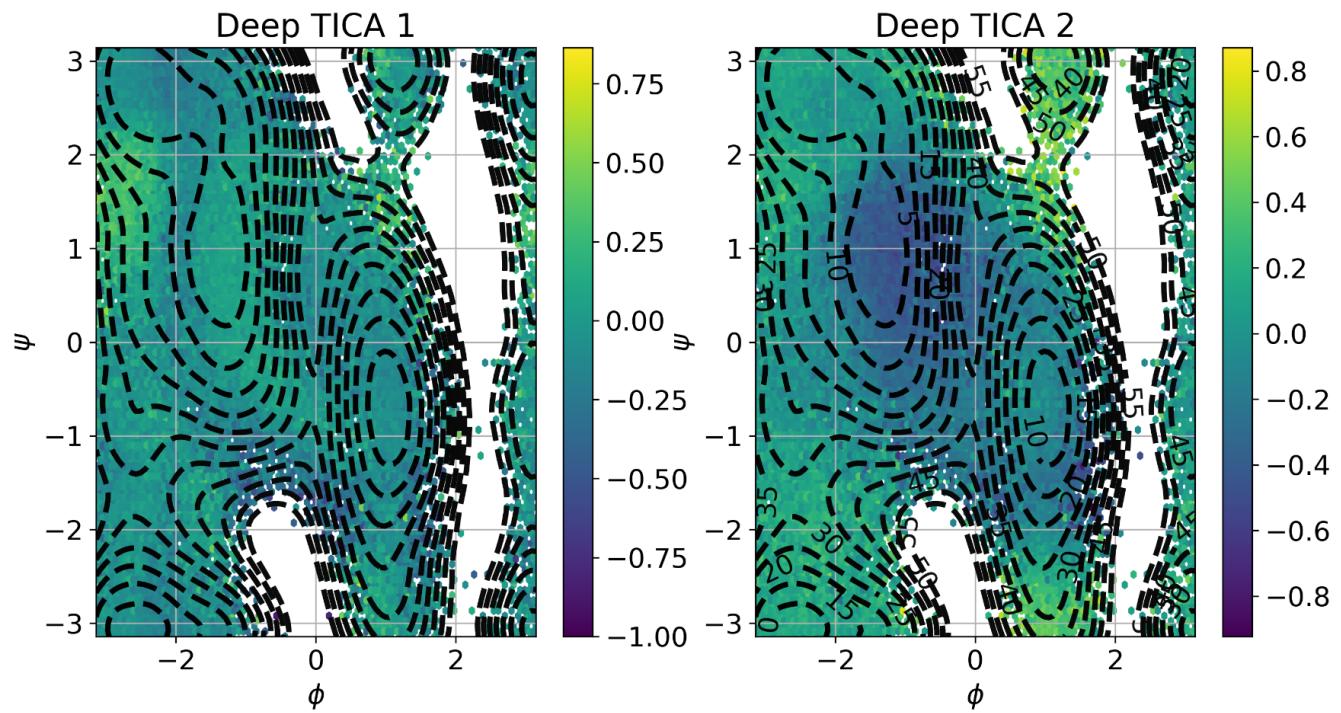
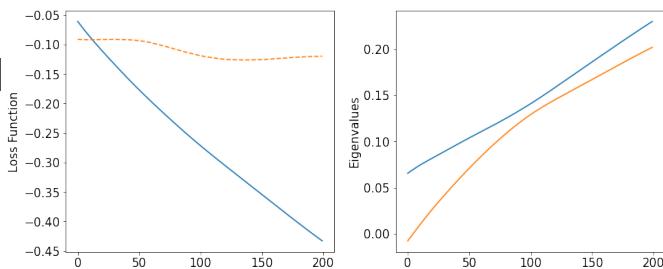
Deep TICA Analysis, without using the rescaled time, lag time = 1



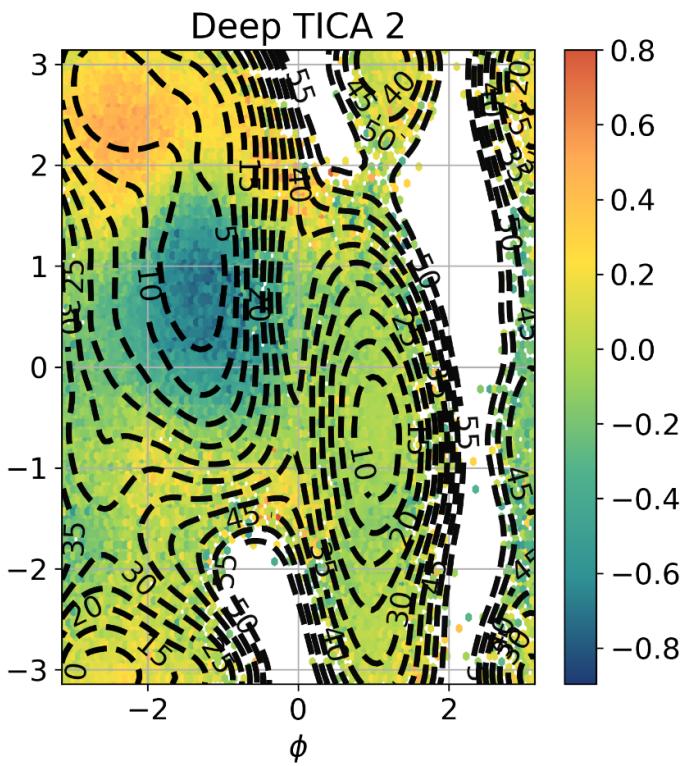
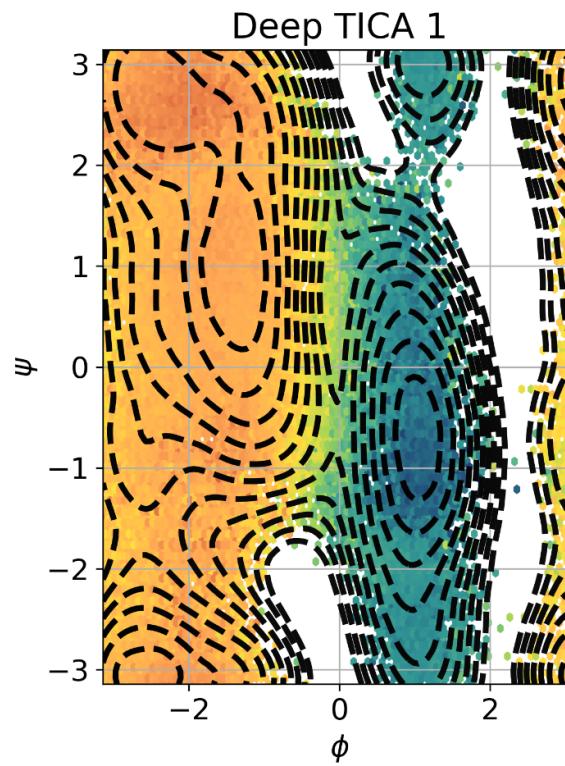
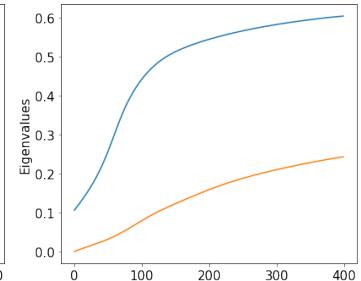
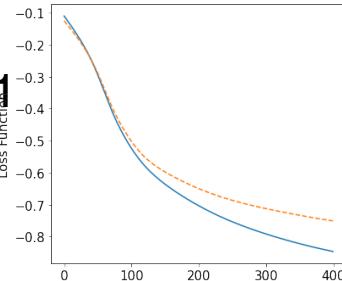
Deep TICA Analysis, rescaled as option (1) ,lag time = 25



Deep TICA Analysis, rescaled as option (2) ,lag time = 1



Deep TICA Analysis, rescaled as option (3) ,lag time = 1



TICA Analysis

In order to detect theoretical problems that can be hidden by the Neural Network architecture I decide to perform a simple TICA Analysis, using the same descriptors and data.

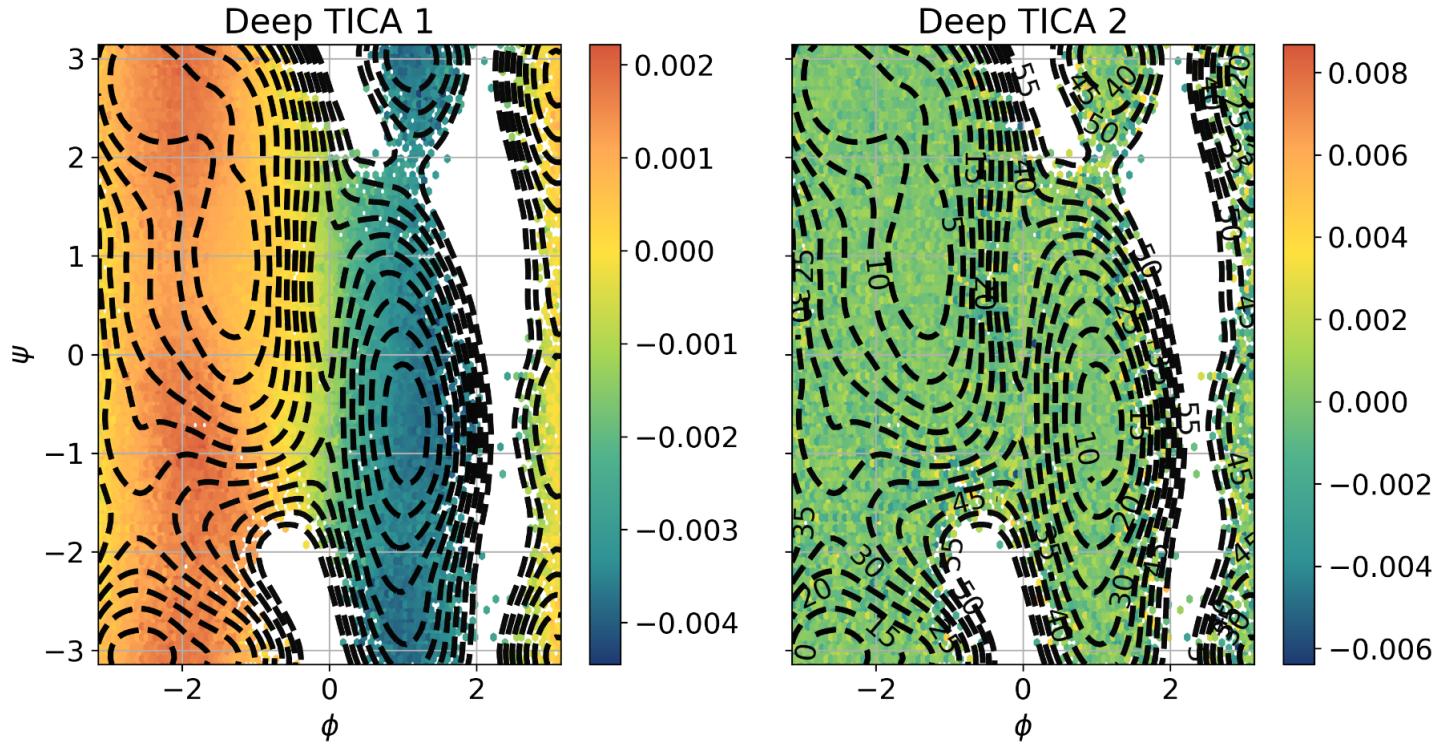
The lag time for this particular case has been also chosen to not have mathematical issues with the diagonalization of the TICA matrix ($m = C(\tau) \cdot C^{-1}(0)$) when couples of data are collected using the rescaled lag time.

I specifically met this kind of issue for the rescaling option (1) and (2)

TICA Analysis, without using the rescaled time, lag time

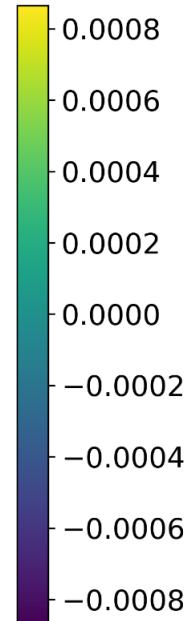
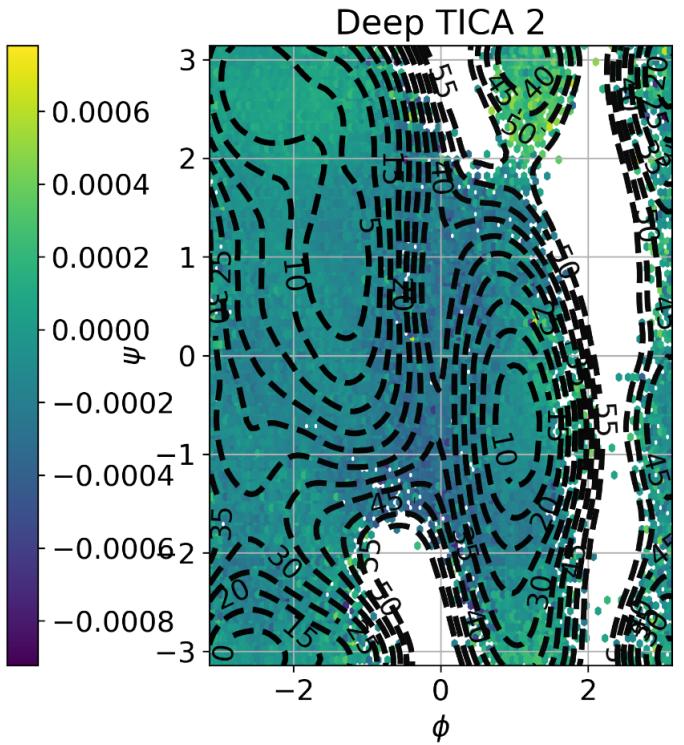
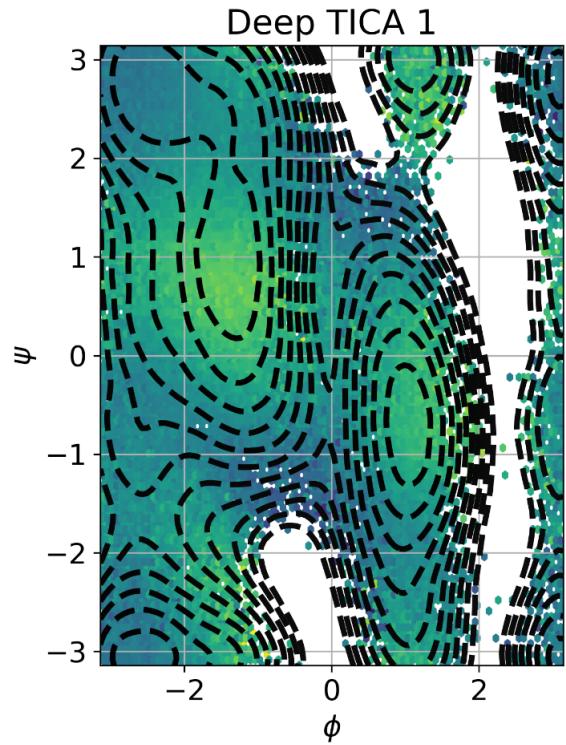
= 1

$\lambda_1 = 0.683$



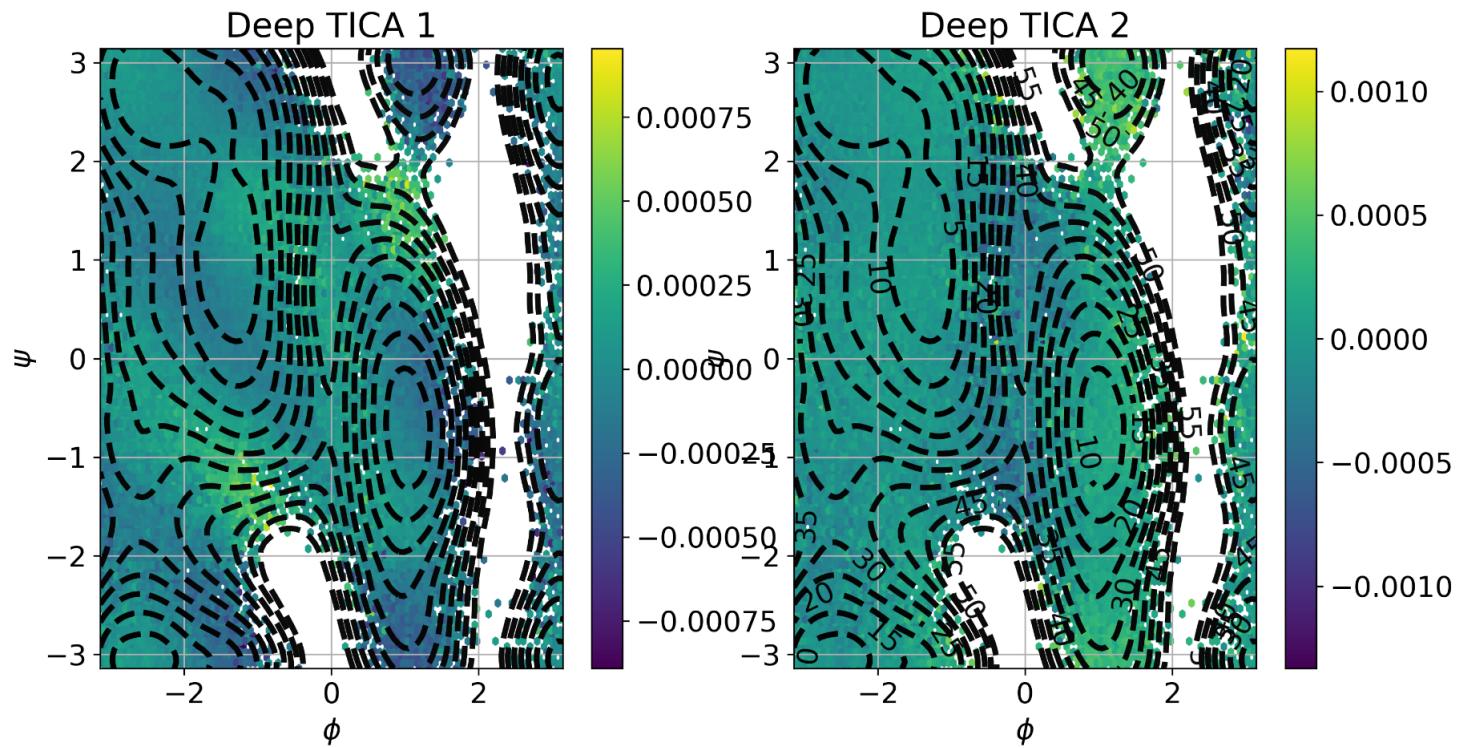
TICA Analysis, rescaled as option (1) ,lag time = 20

$$\lambda_1 = 0.23$$



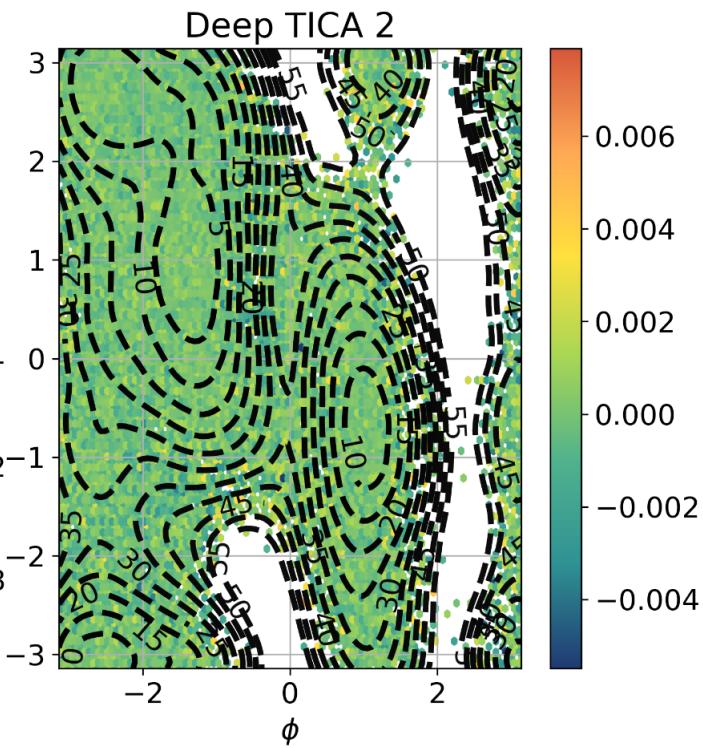
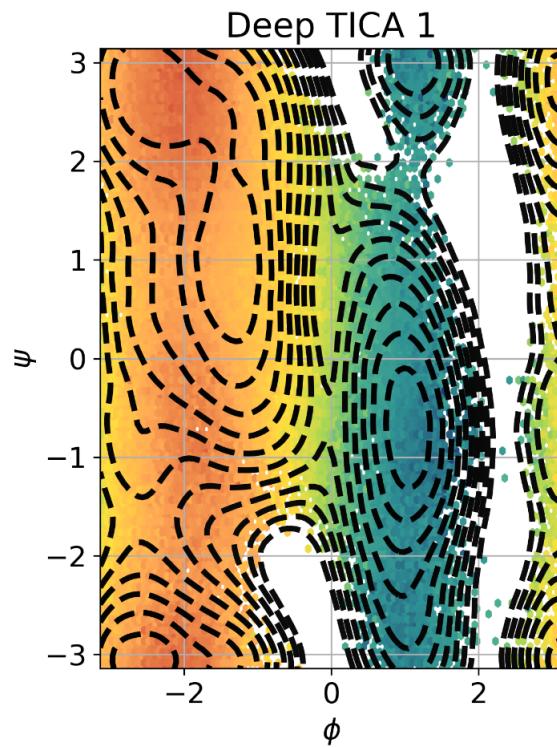
TICA Analysis, rescaled as option (2) ,lag time = 0.01

$$\lambda_1 = 1.96$$



TICA Analysis, rescaled as option (3) ,lag time = 1

$$\lambda_1 = 0.59$$



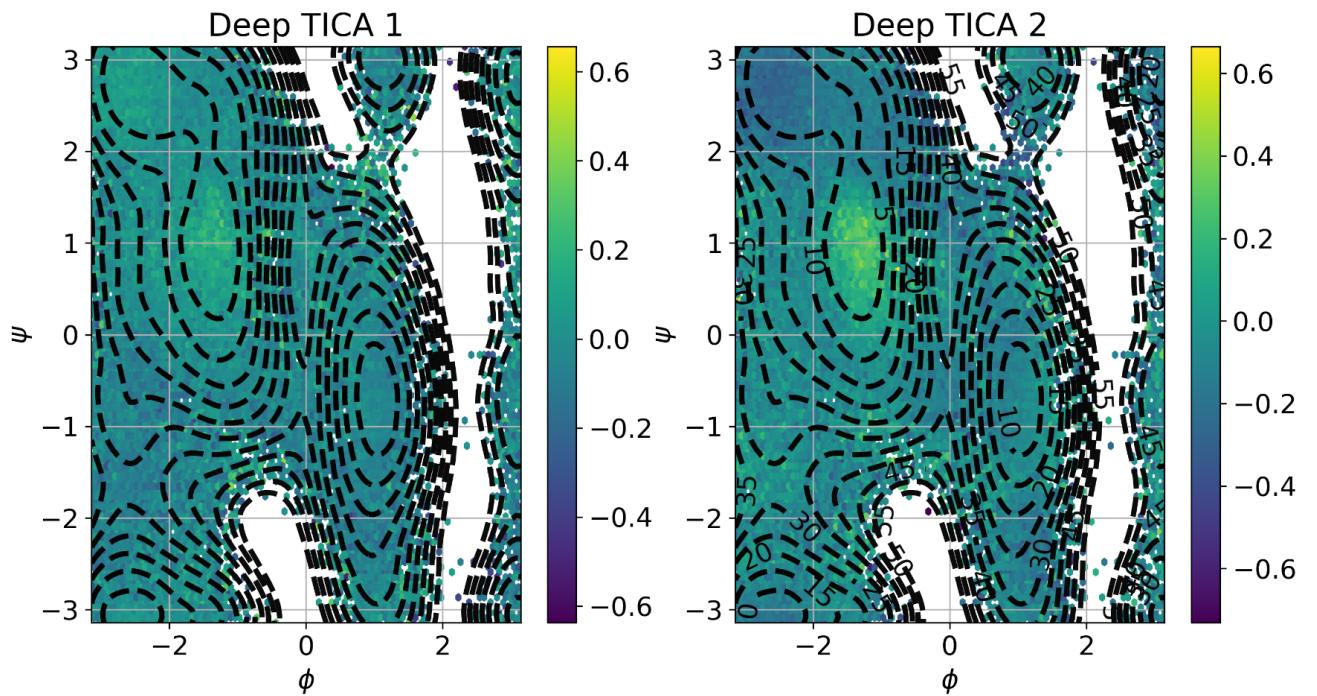
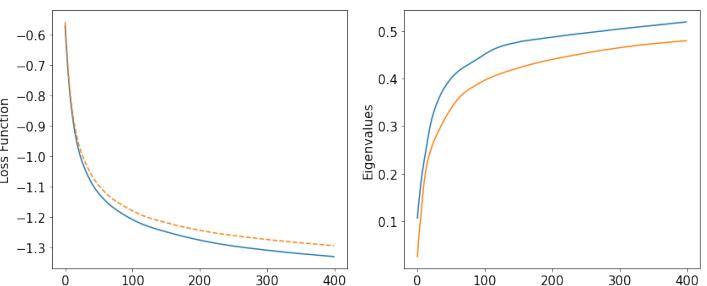
Comments on cases (1) and (2)

The trend of the loss function during training suggests that the model is overfitting. The overfitting arises from two main different scenarios:

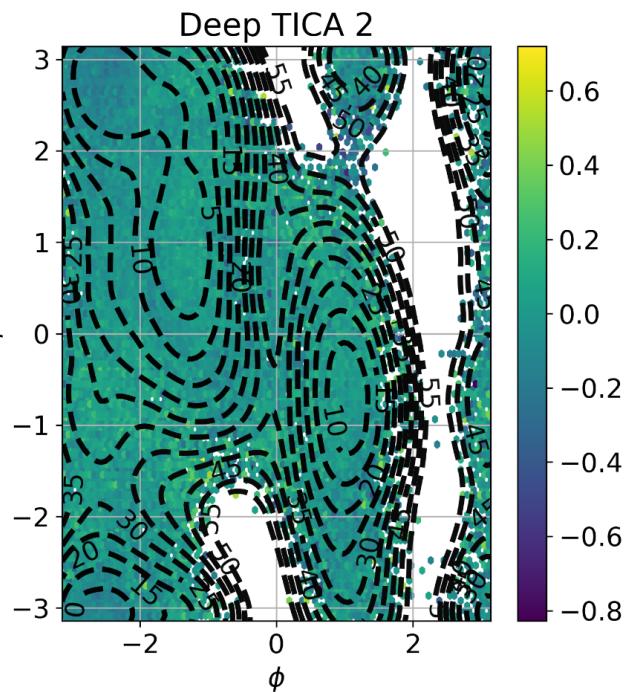
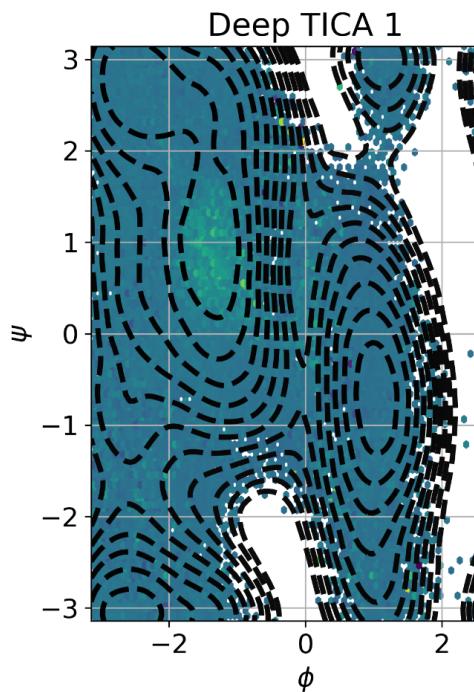
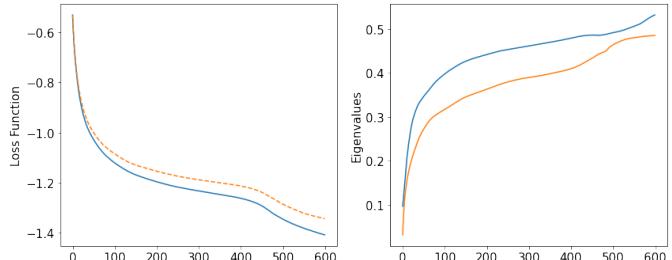
1. The model is too much complicated, so for instance one can try to change the hyperparameters, including the choice of the lag time value. I've tried to change them but without success (random trials).
2. The trend of the validation loss function suggests another scenario: the data are affected by a lot of noise (another possibility is that the two sets, training and validation, come from different distribution, but I discard this option). Basically all the data are noisy and no signal is detected during the training.

One possibility to improve the trend of the validation loss function is to train the model with batches and each batch contains correlated couples at different lag times. If some batches contain a signal then the model will be able to detect it.

Deep TICA Analysis, rescaled as option (1),lag time = 0.01,25 and 50 batches



Deep TICA Analysis, rescaled as option (2), lag time = 0.001, 1 and 50 batches



Questions

1. Why the Deep TICA Analysis is able to detect an acceptable approximation of the slow modes using the not rescaled time?
2. Why the Deep TICA Analysis is not able to detect an acceptable approximation of the slow modes using the rescaled time? This latter question can be rearranged this way: What is the origin of the noise in training data obtained using the rescaled time?

I recall that the time difference between two consecutive configurations depends only on the bias value in the last configuration

$$\Delta t'_{i+1,i} = \Delta t \left(\sum_{k=0}^{i+1} e^{\beta V_k} - \sum_{k=0}^i e^{\beta V_k} \right) = \Delta t \cdot e^{\beta V_{i+1}}$$

It's easier to answer the second question. Looking at the distribution of the rescaled time we can see that most of the sampled data are distributed in high energy regions (the corresponding rescaled time is really small). These data are strongly noisy. We hope that the rescaled time approach is in principle able to distinguish between slow motions (data inside the basin) and fast motions (high energy regions). But apparently it is not. The algorithm fails to find the (approximately) "unbiased" correlated couples.

In other words, with this algorithm when a lot of noise (diffusive dynamics) is present we are not able to reconstruct the unbiased distribution (Boltzmann) of the data.

Why the Deep-TICA Analysis is able to distinguish the slow modes of the system when not using the rescaled time?

My answer is: **Well-tempered target distribution**

The Well-tempered metadynamics (or OPES with target well-tempered distribution) aims to fill the original Free Energy Surface (FES) smoothly, rather than in a flat way, tuning the Gamma parameter.

Metadynamics:

$$F(s) = -V(s)$$

Well-tempered Metadynamics:

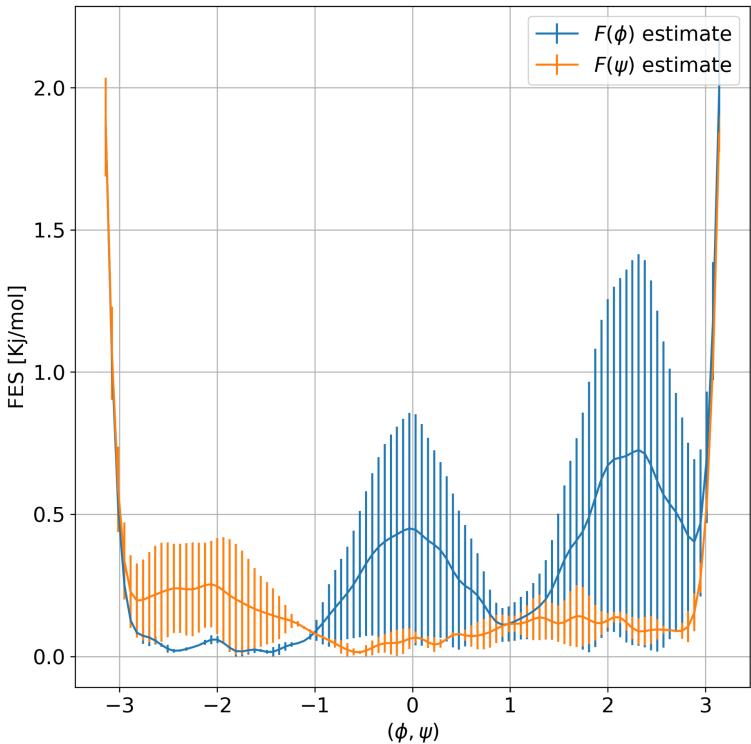
$$F(s) = -\frac{\gamma}{\gamma - 1} V(s)$$

OPES:

$$P^{WT} \propto [P(s)]^{\frac{1}{\gamma}}$$

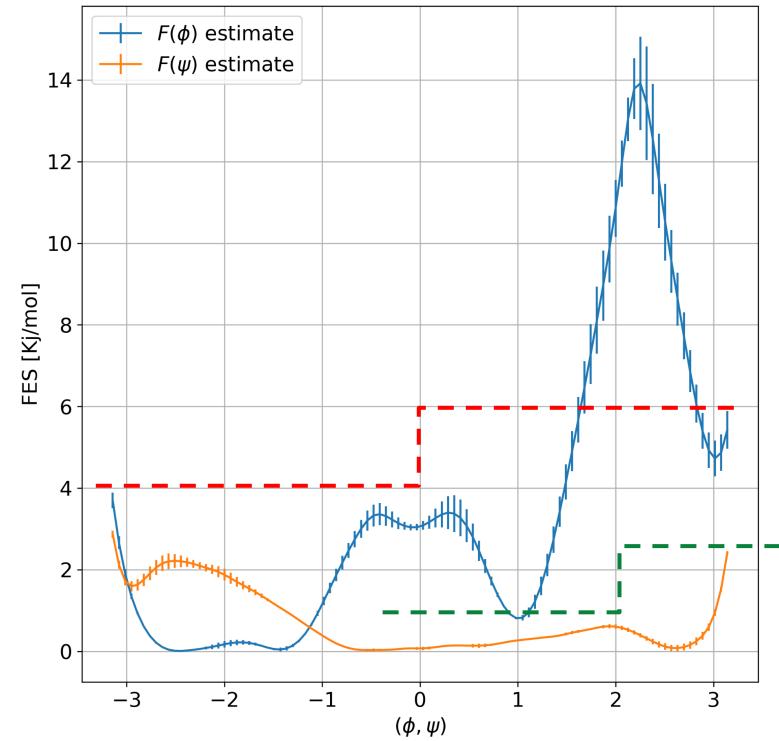
Free Energy not reweighted

Metadynamics



Almost flat

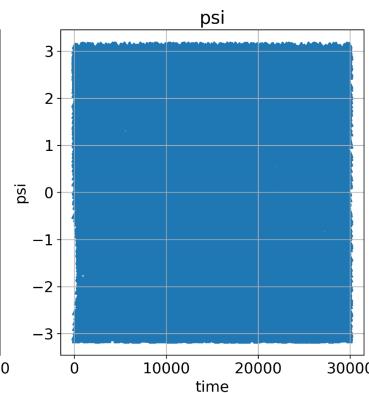
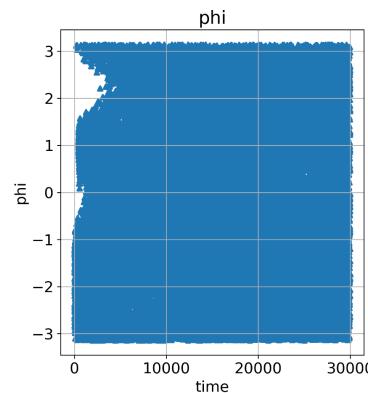
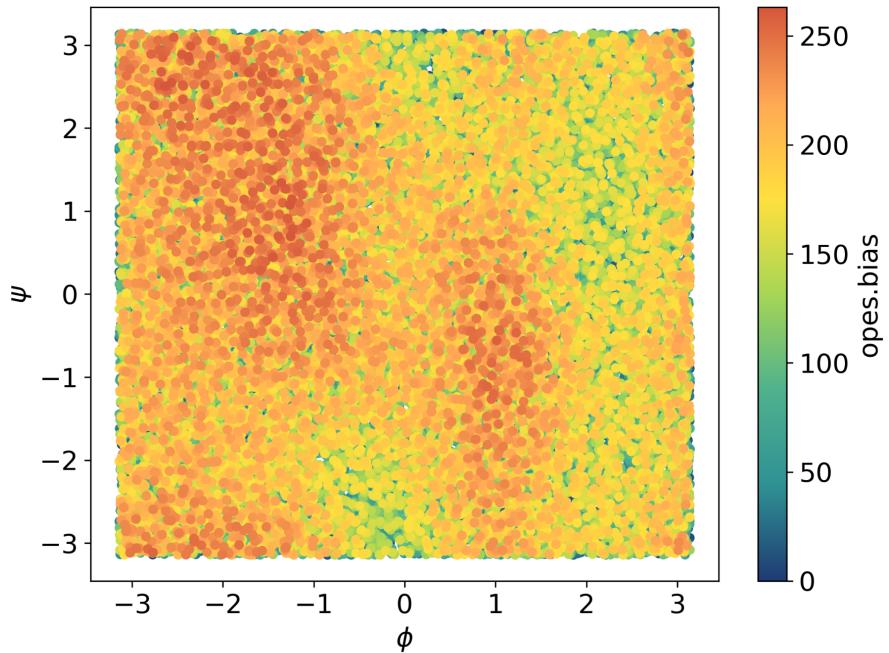
OPES (gamma = 20)



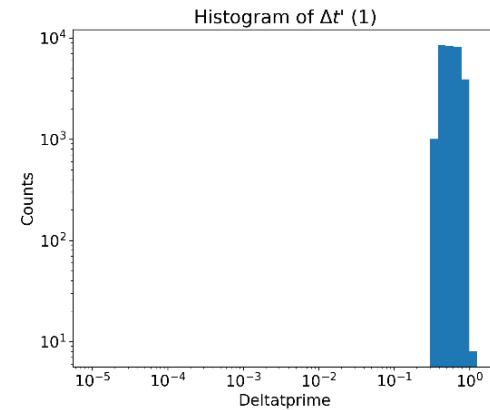
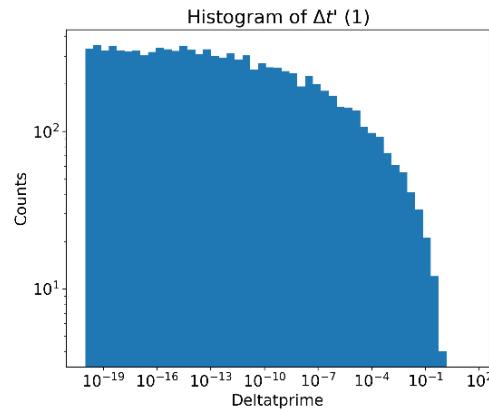
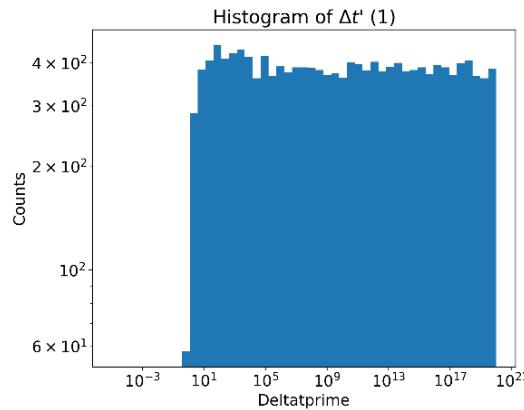
Deep-TICA 1

Deep-TICA 2

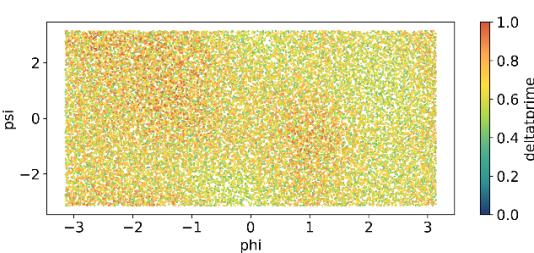
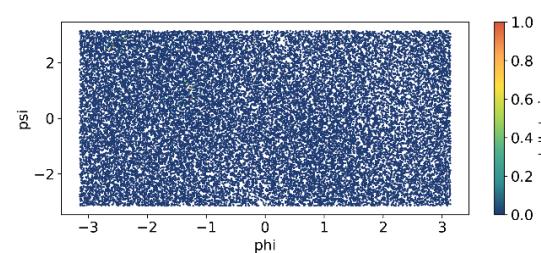
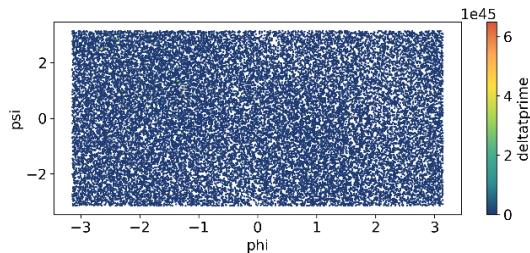
Metadynamics simulation, 30ns



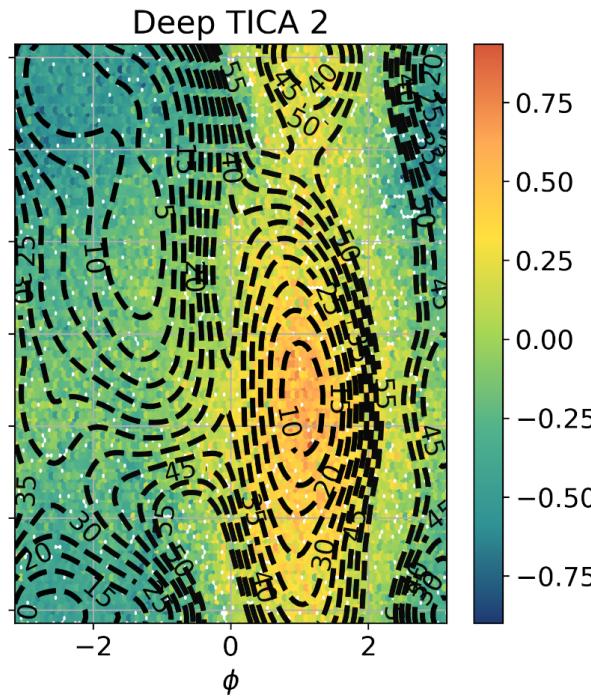
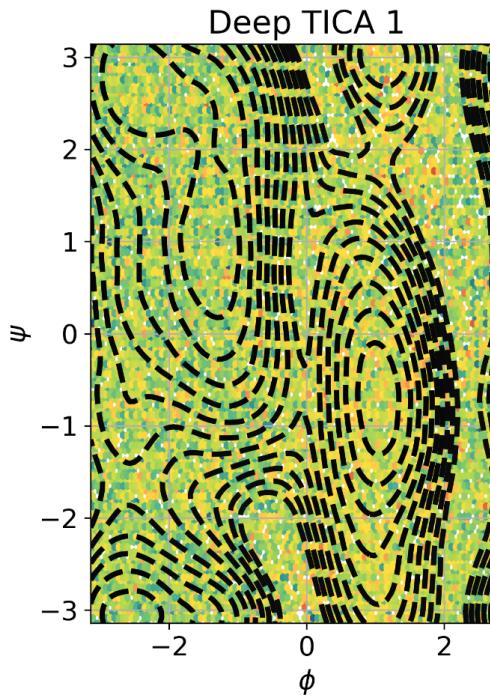
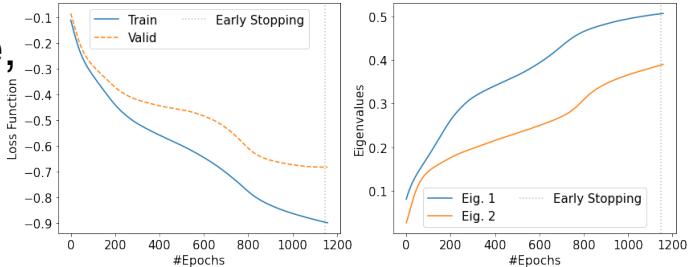
Distribution of Delta tprime



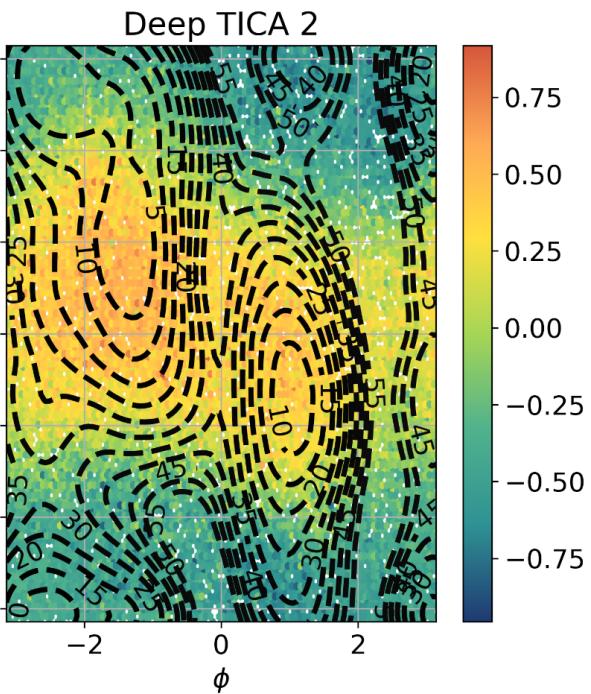
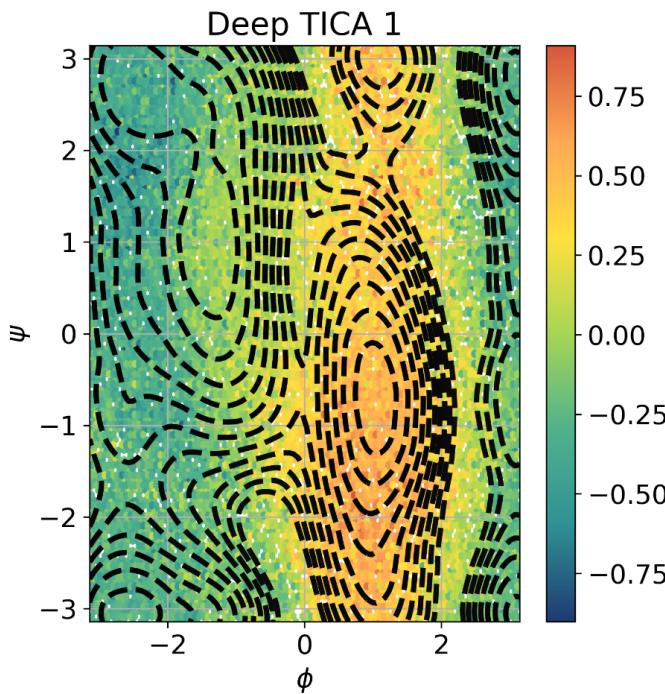
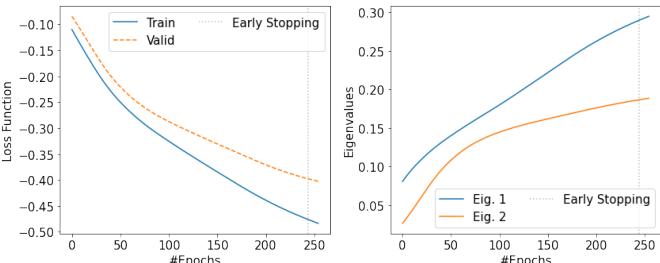
Trajectories



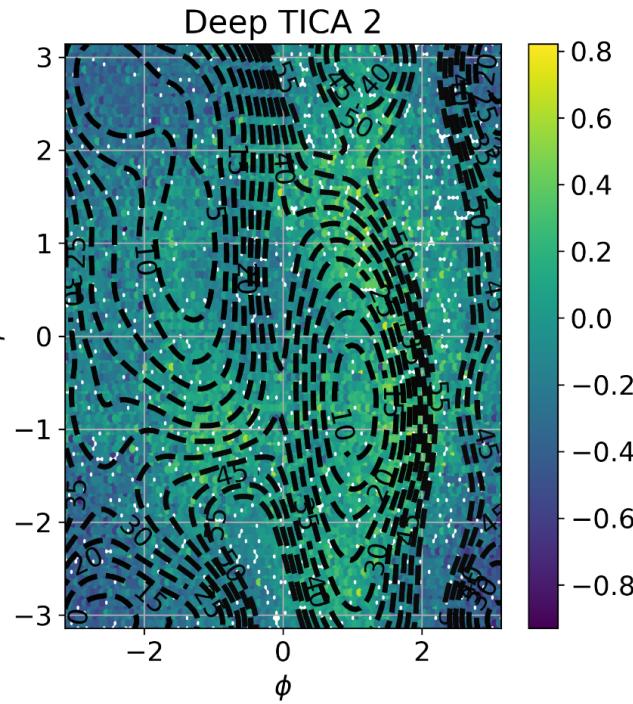
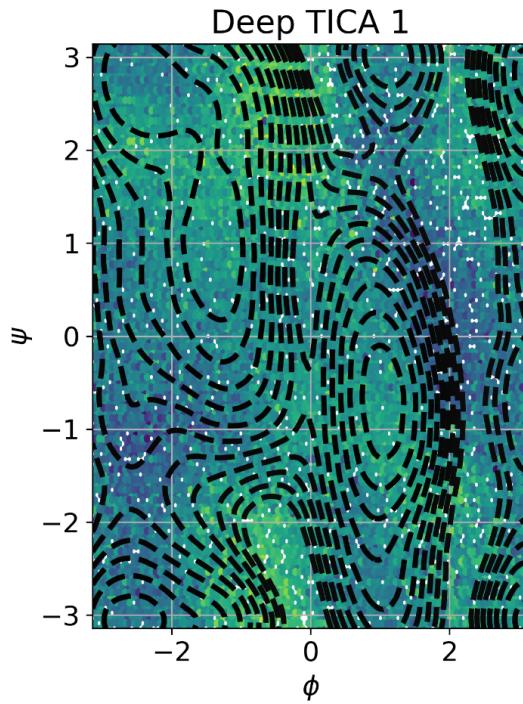
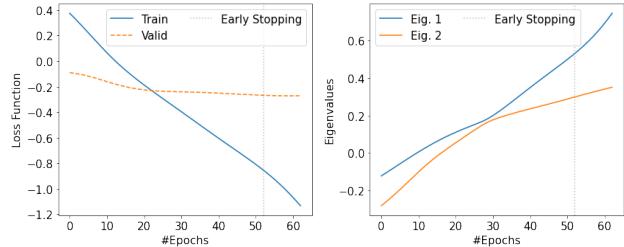
Deep TICA Analysis, without using the rescaled time, lag time = 1



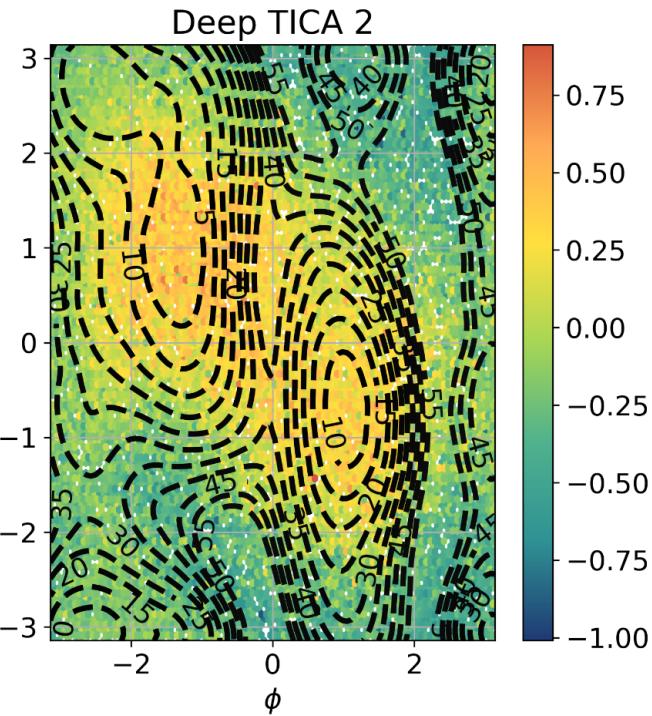
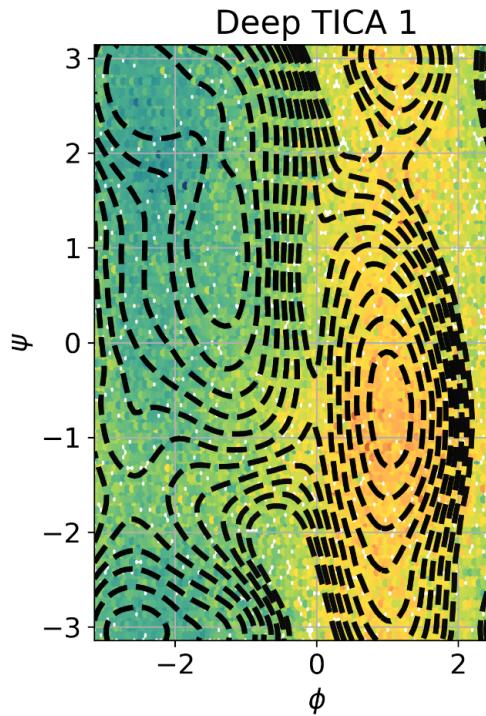
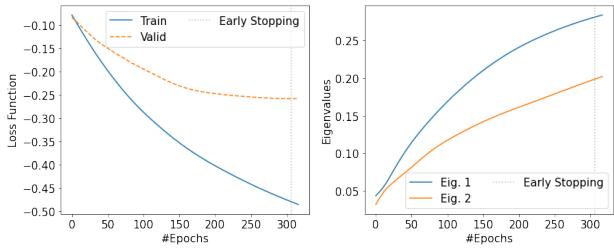
Setting an Early Stopping that checks if the validation loss do not decrease too much ($\delta = 0.005$)



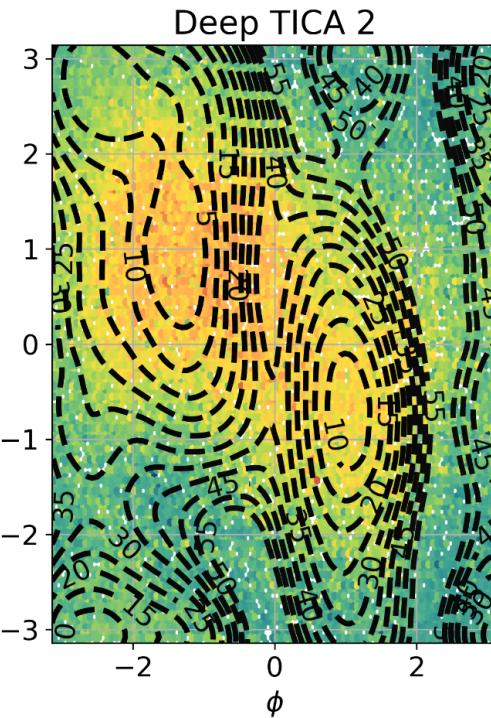
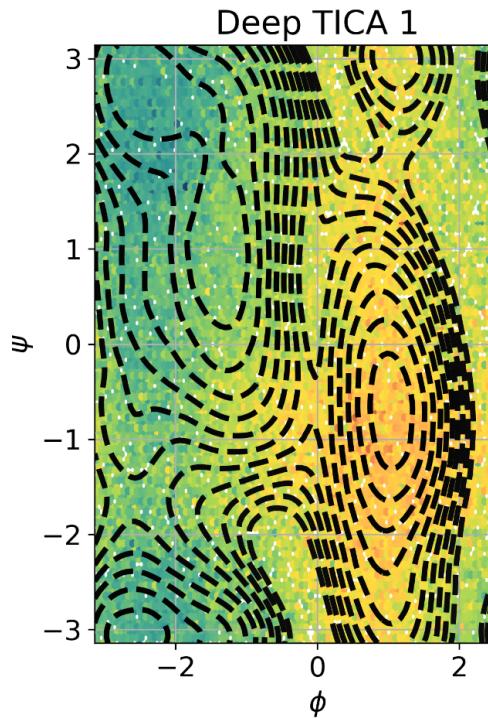
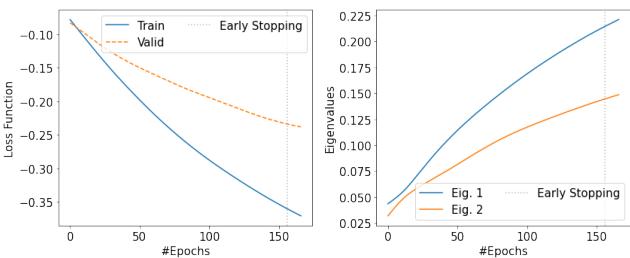
Deep TICA Analysis, using the rescaled time option (2), lag time = 1



Deep TICA Analysis, using the rescaled time option (3), lag time = 1



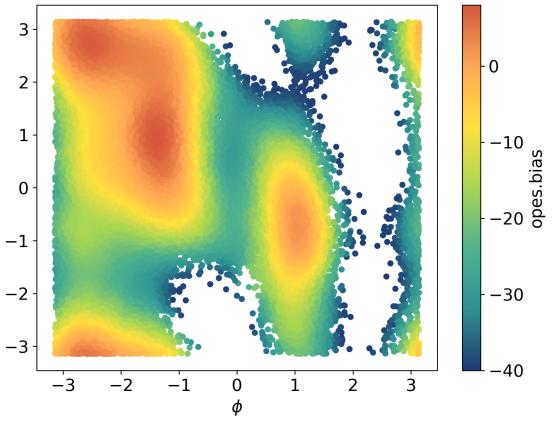
Setting an Early Stopping that checks if the validation loss do not decrease too much (delta = 0.005)



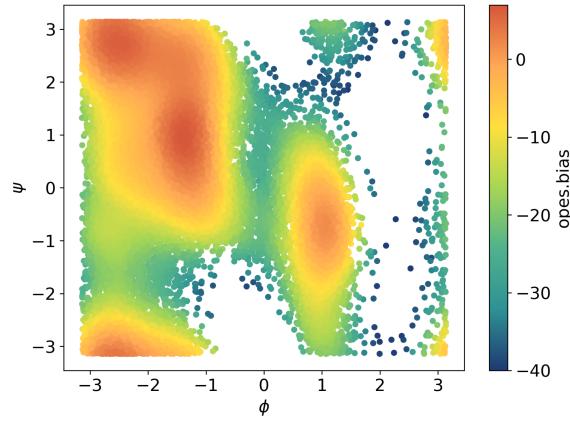
What does it happen if we decrease Gamma ?

The exploration of the FES can be limited to the physically interesting regions.

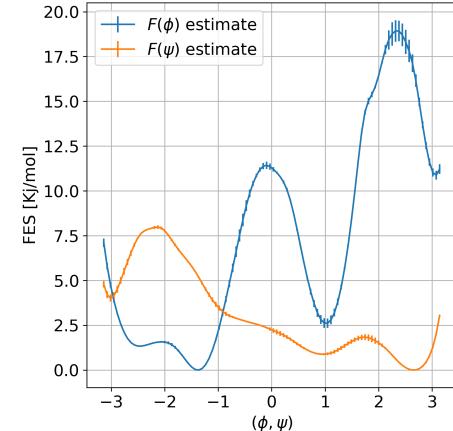
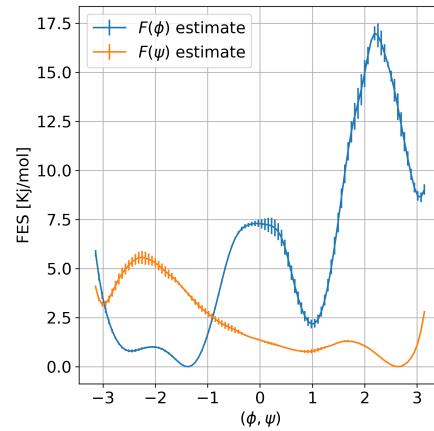
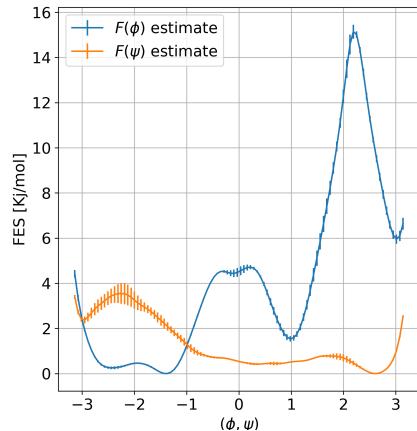
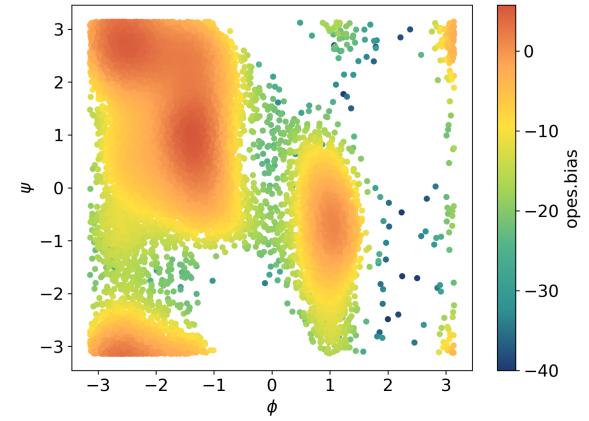
$$\gamma = 10$$



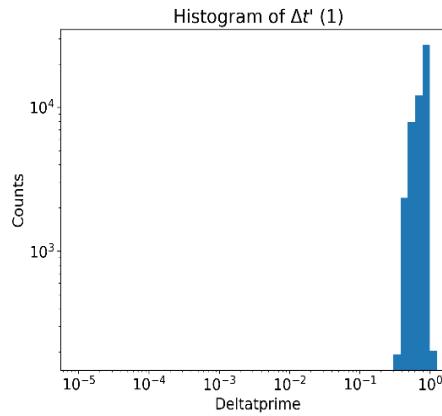
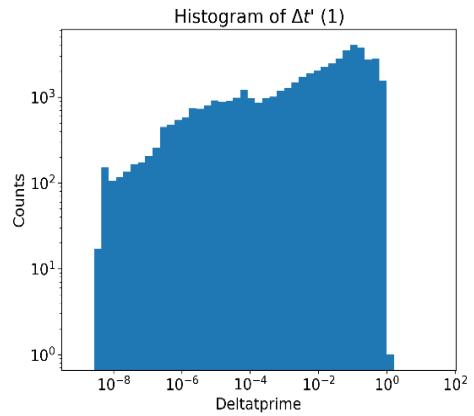
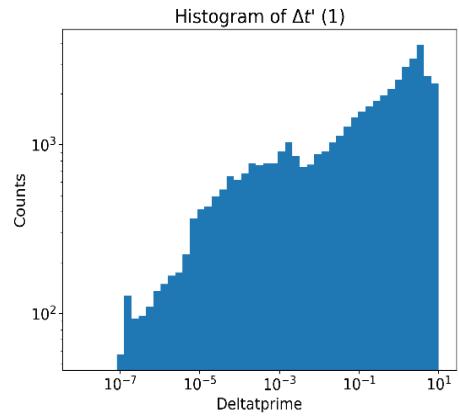
$$\gamma = 5$$



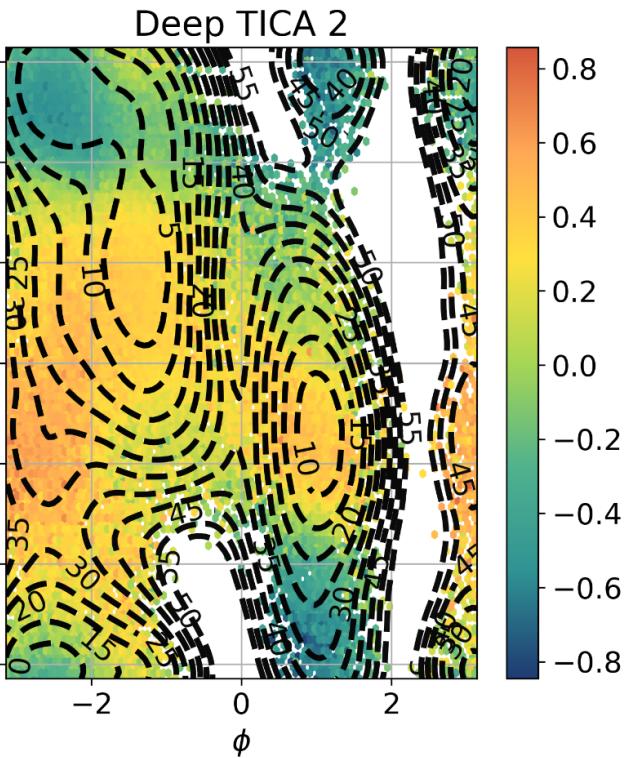
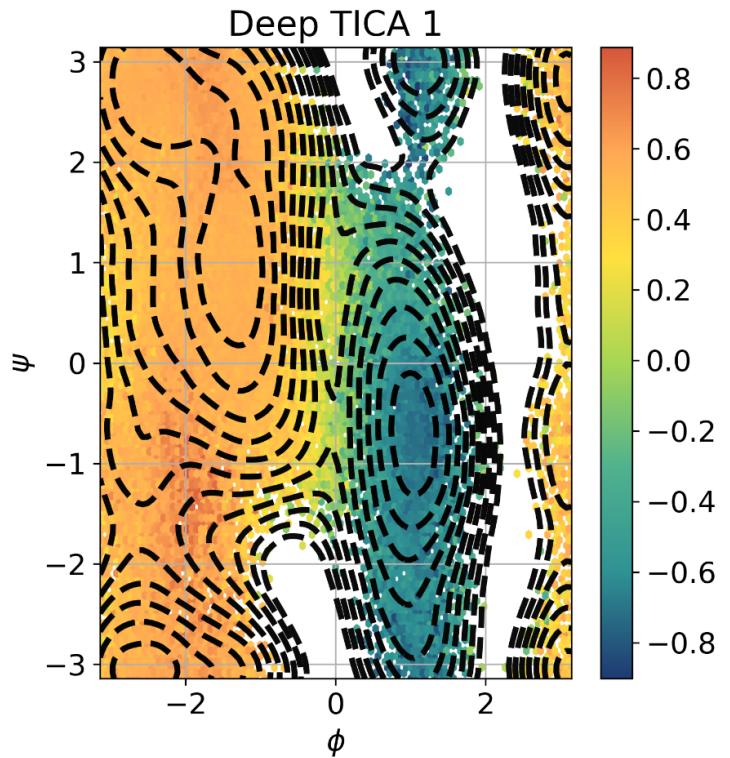
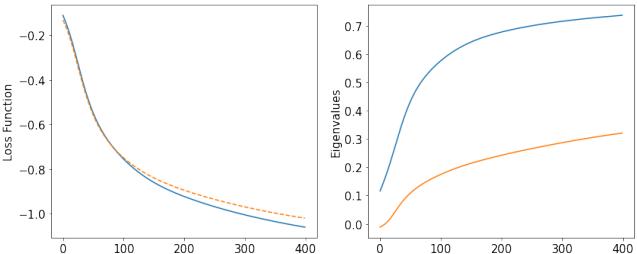
$$\gamma = 3$$



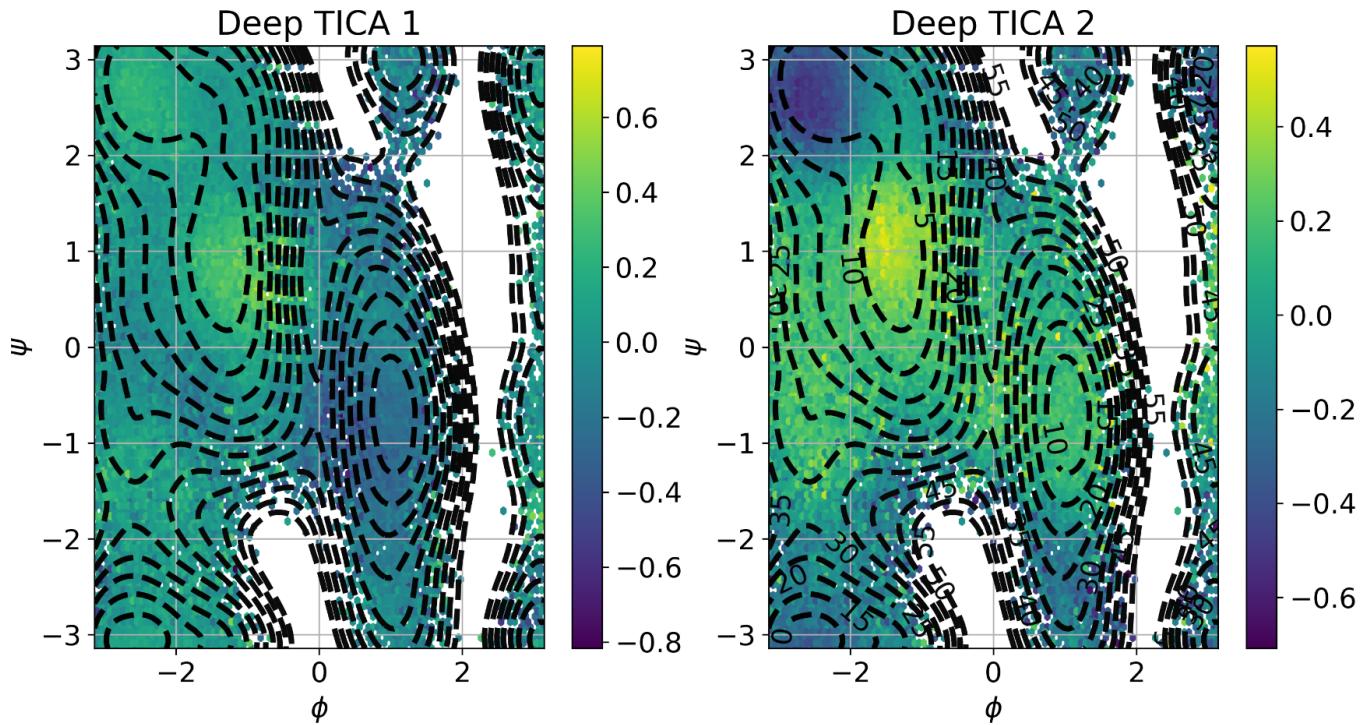
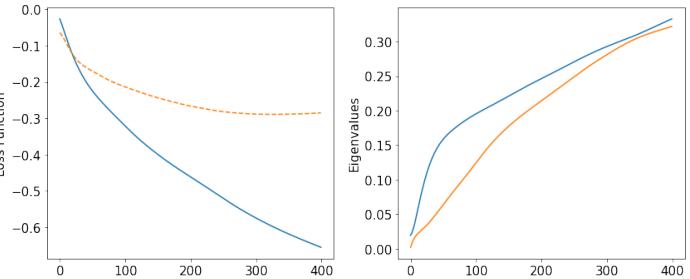
$$\gamma = 10$$



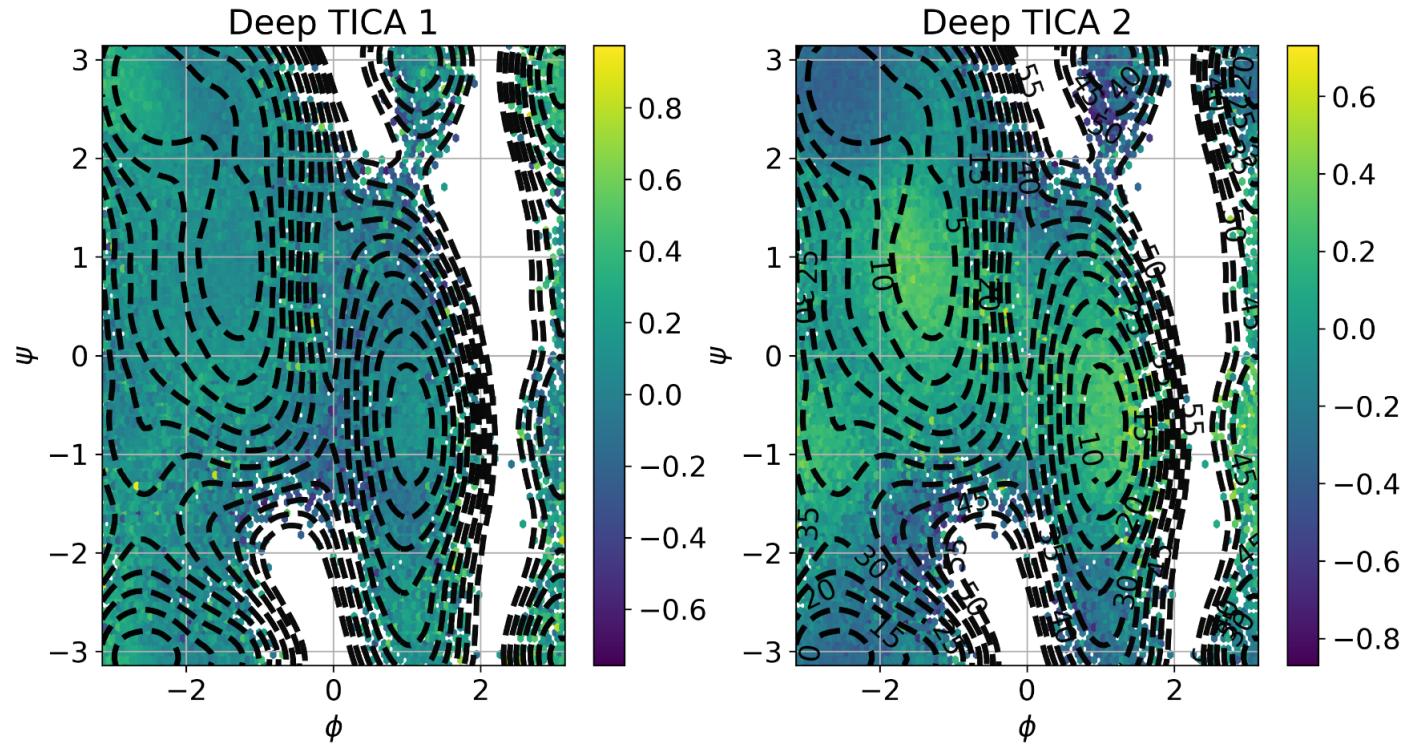
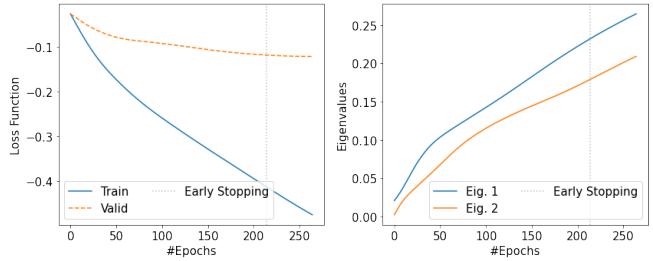
Deep TICA Analysis, without using the rescaled time, lag time = 1



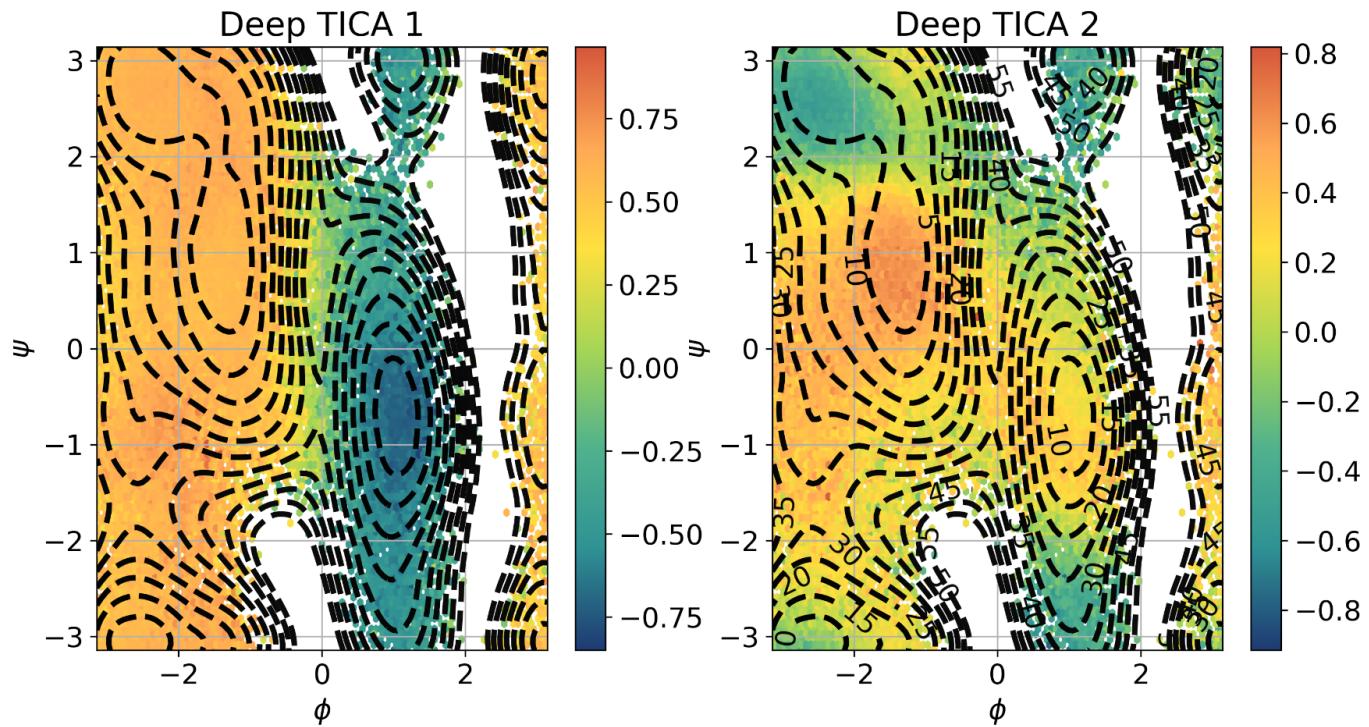
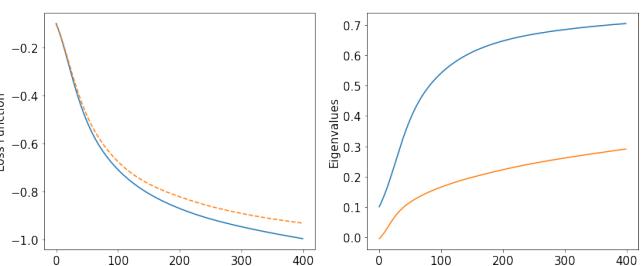
Deep TICA Analysis, rescaled as option (1) ,lag time = 15



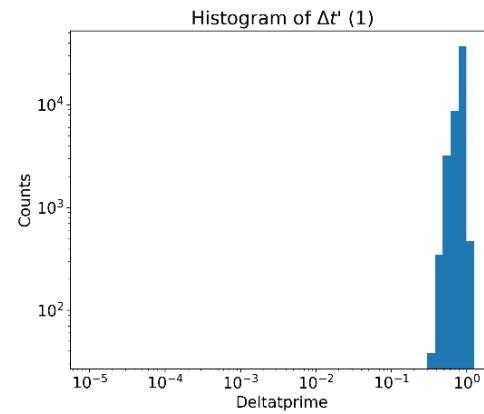
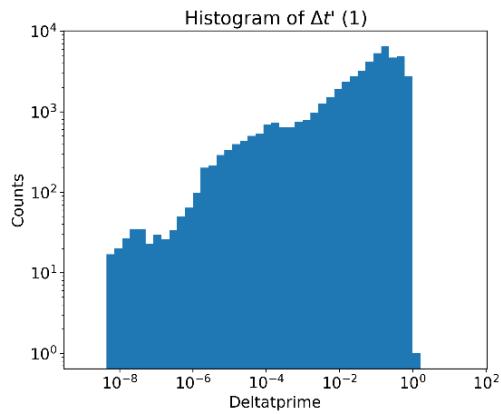
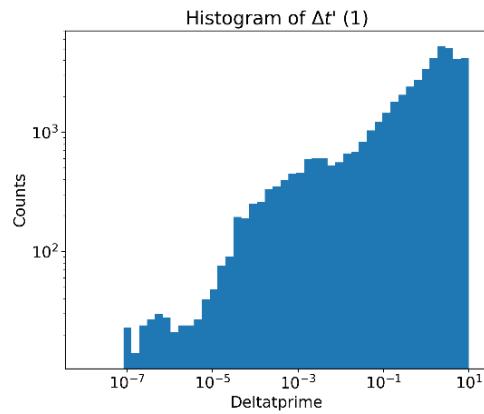
Deep TICA Analysis, rescaled as option (2) ,lag time = 1



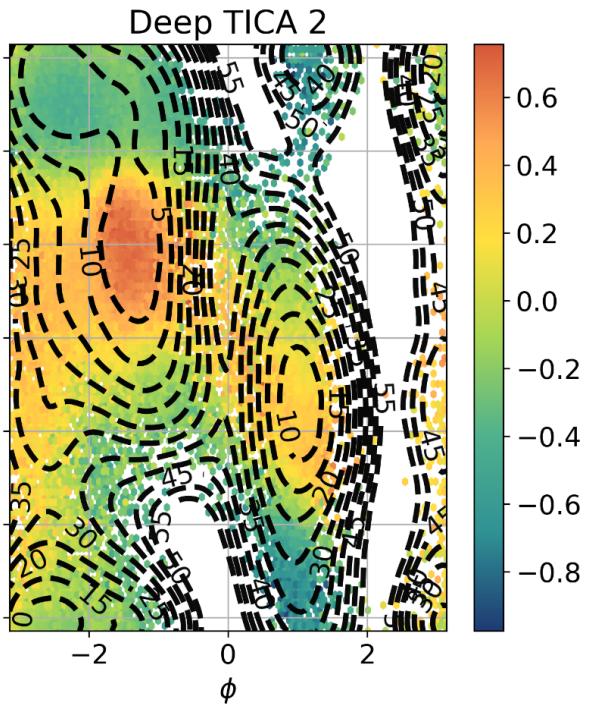
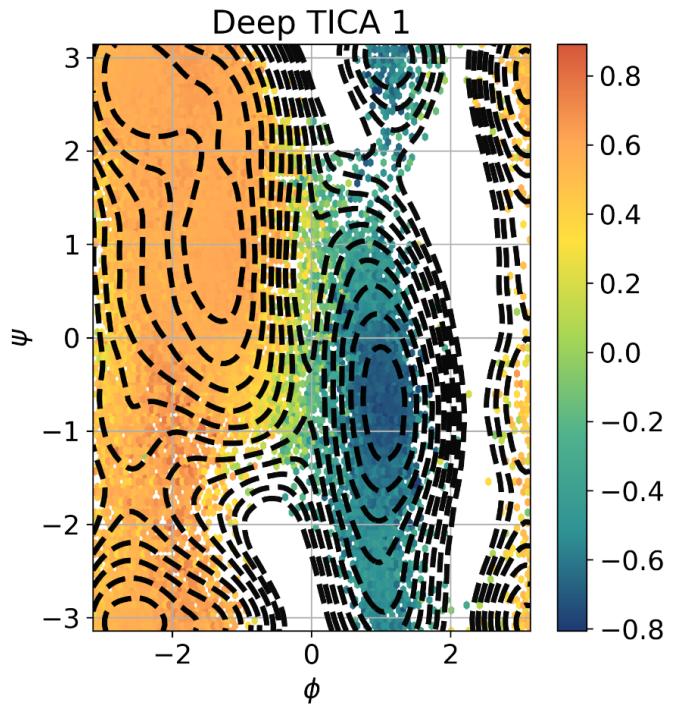
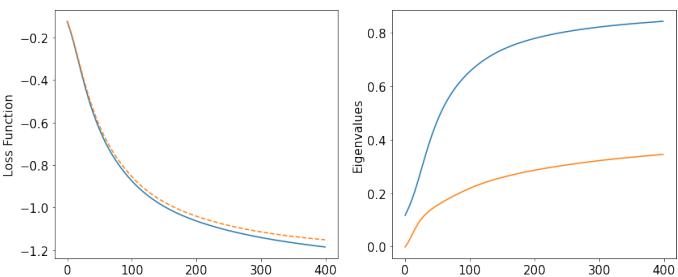
Deep TICA Analysis, rescaled as option (3) ,lag time = 1



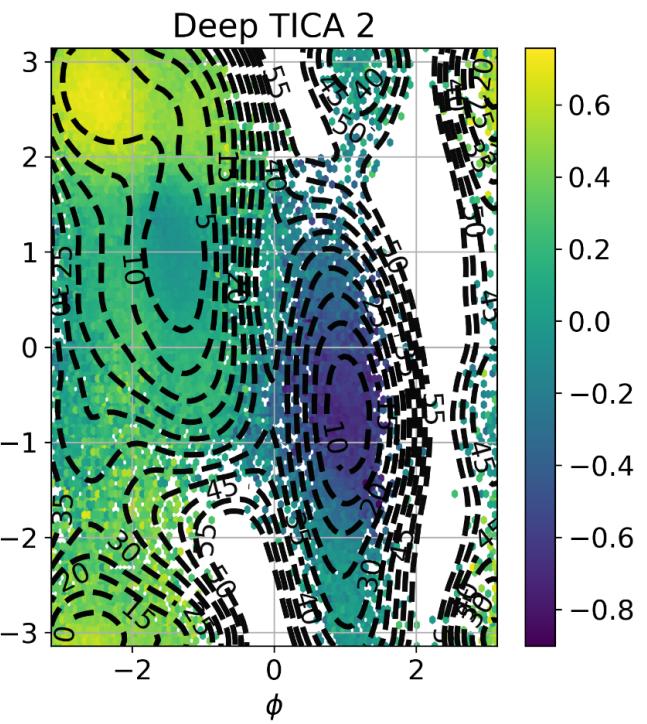
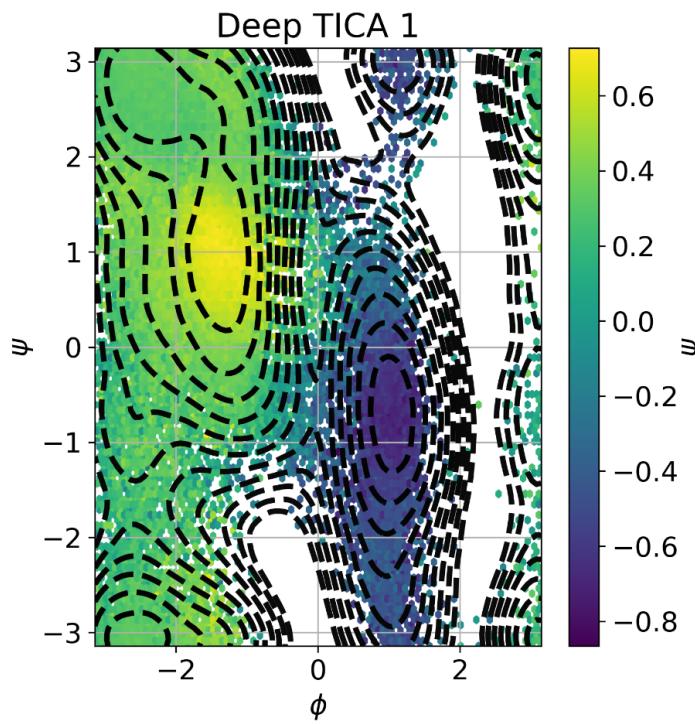
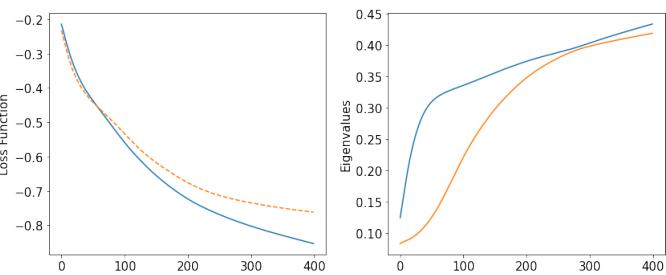
$$\gamma = 5$$



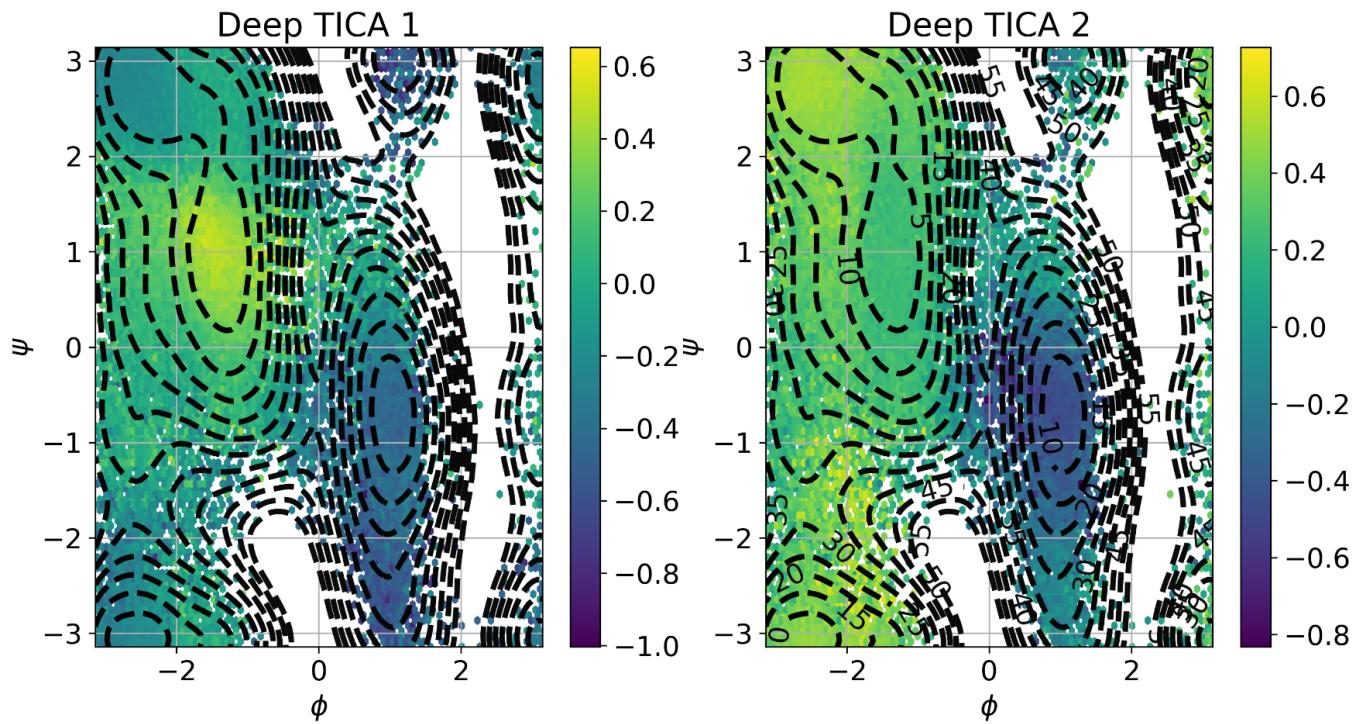
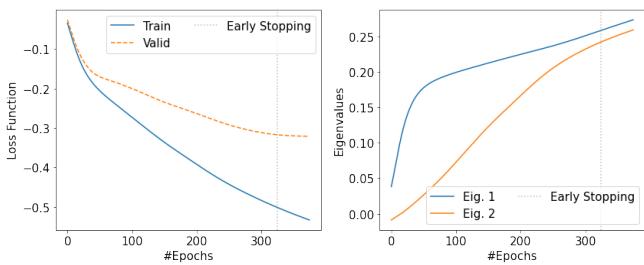
Deep TICA Analysis, without using the rescaled time, lag time = 1



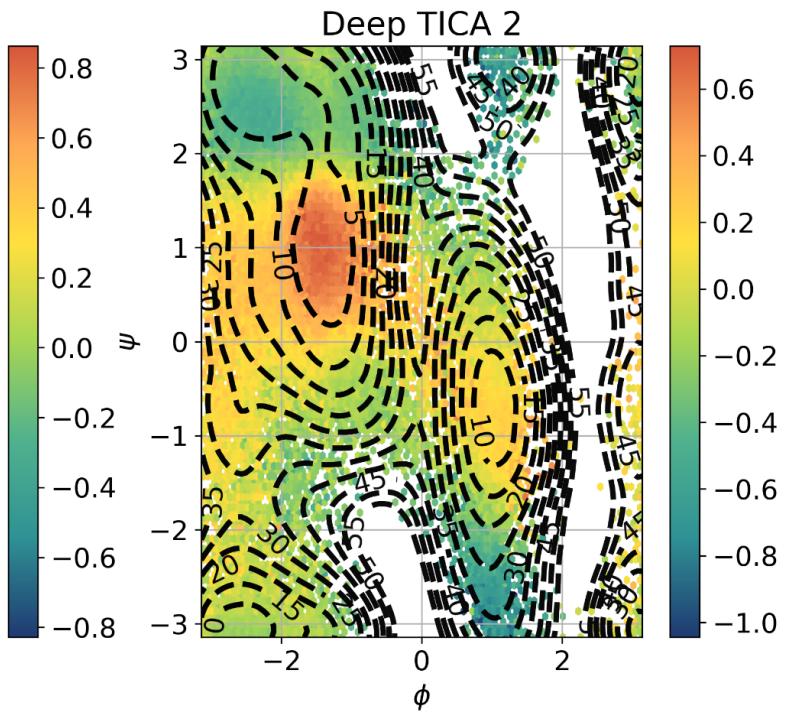
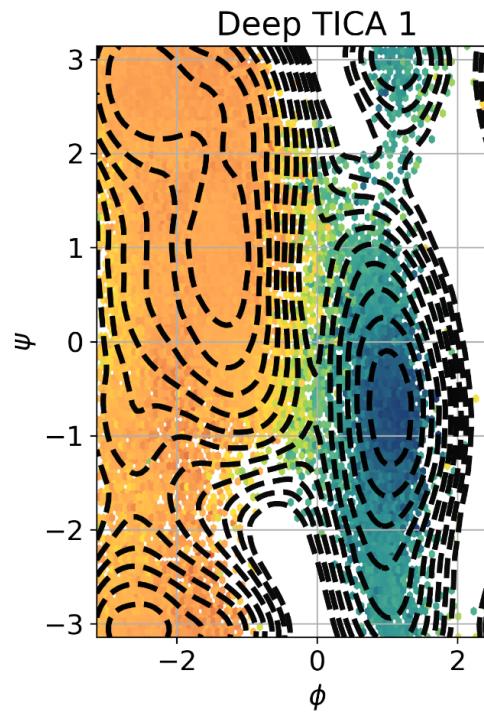
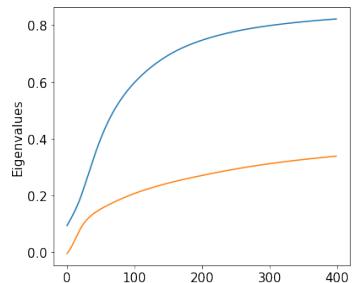
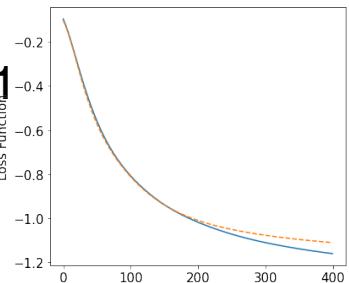
Deep TICA Analysis, rescaled as option (1) ,lag time = 7.5



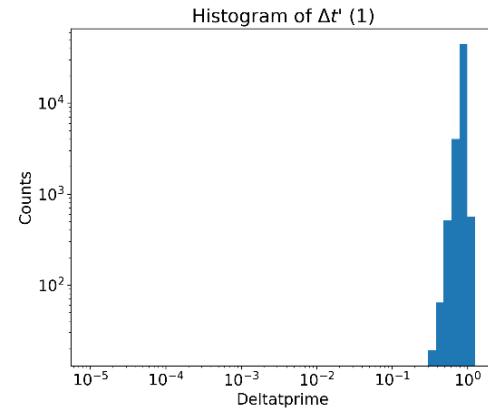
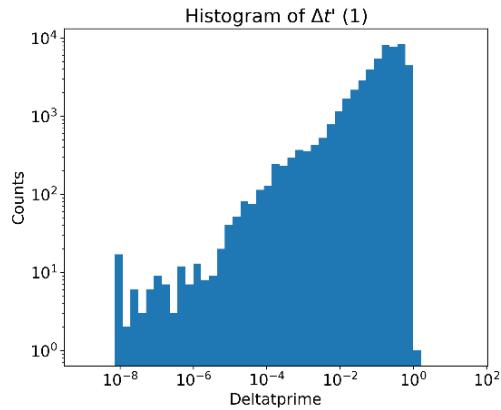
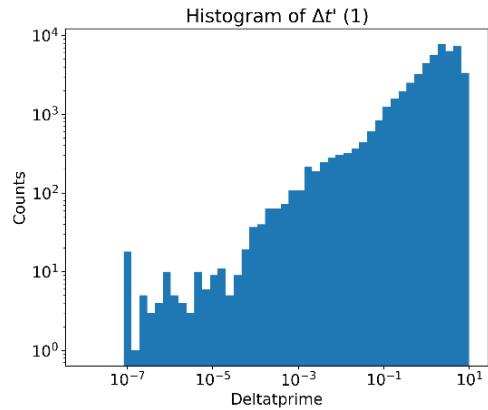
Deep TICA Analysis, rescaled as option (2) ,lag time = 1



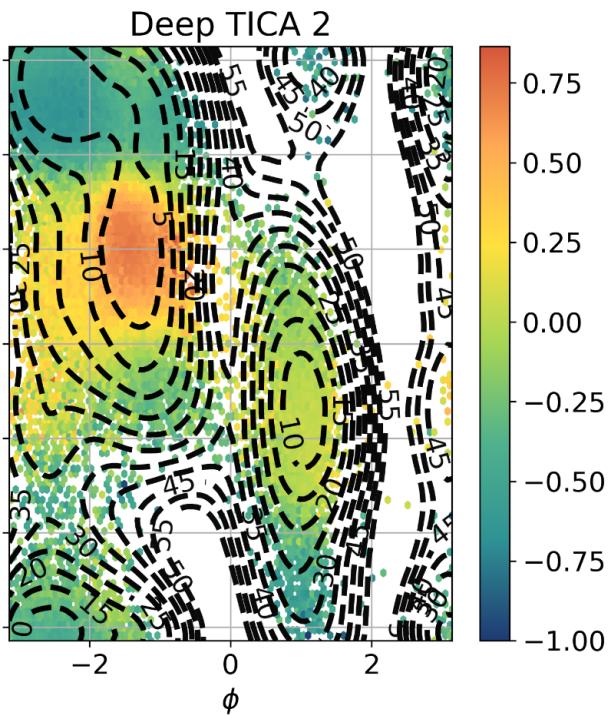
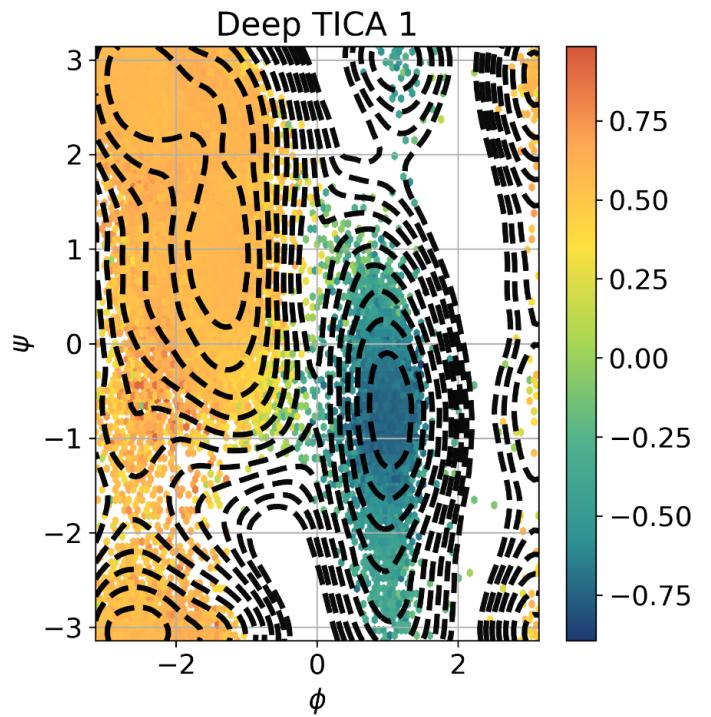
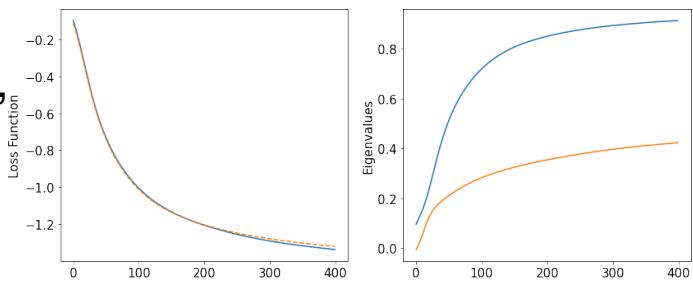
Deep TICA Analysis, rescaled as option (3) ,lag time = 1



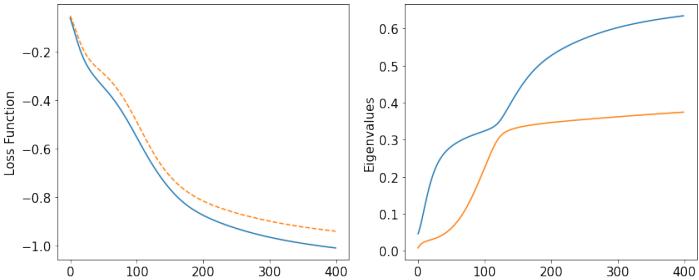
$$\gamma = 3$$



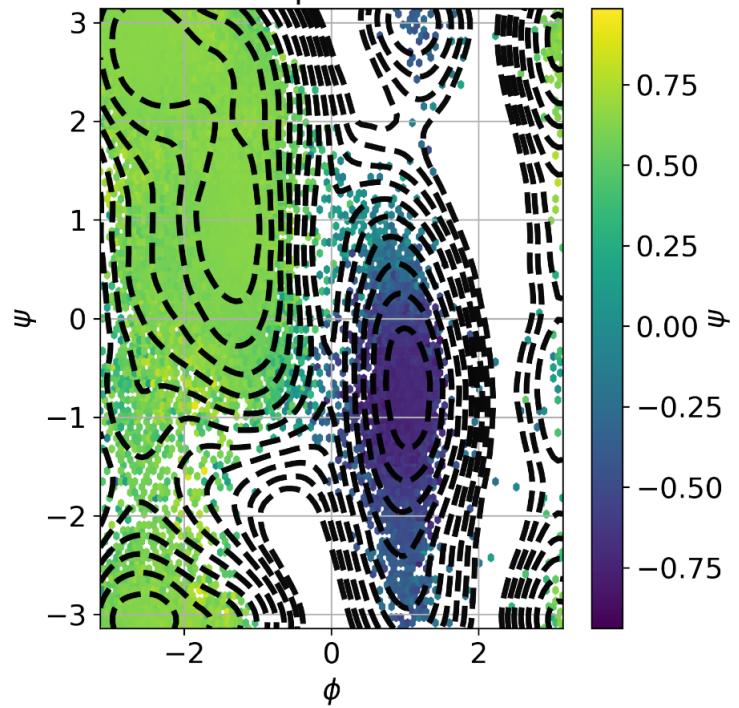
Deep TICA Analysis, without using the rescaled time, lag time = 1



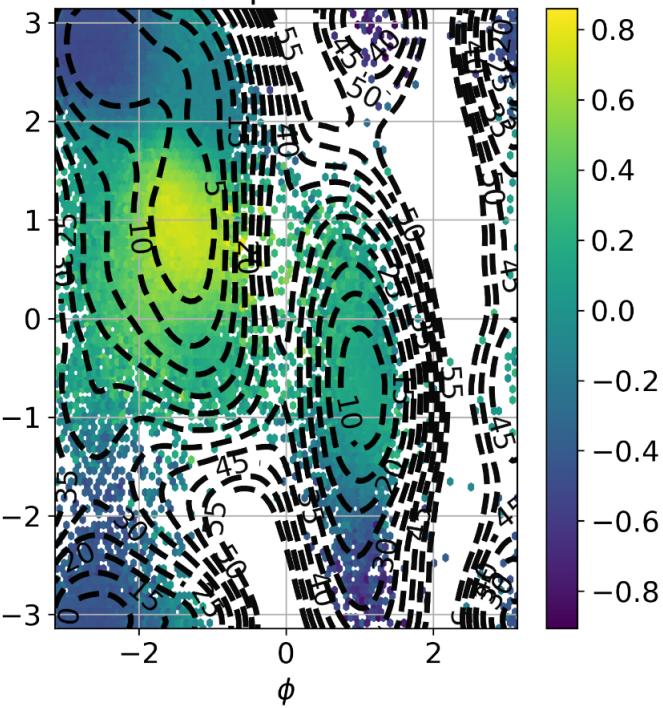
Deep TICA Analysis, rescaled as option (1) ,lag time = 7.5



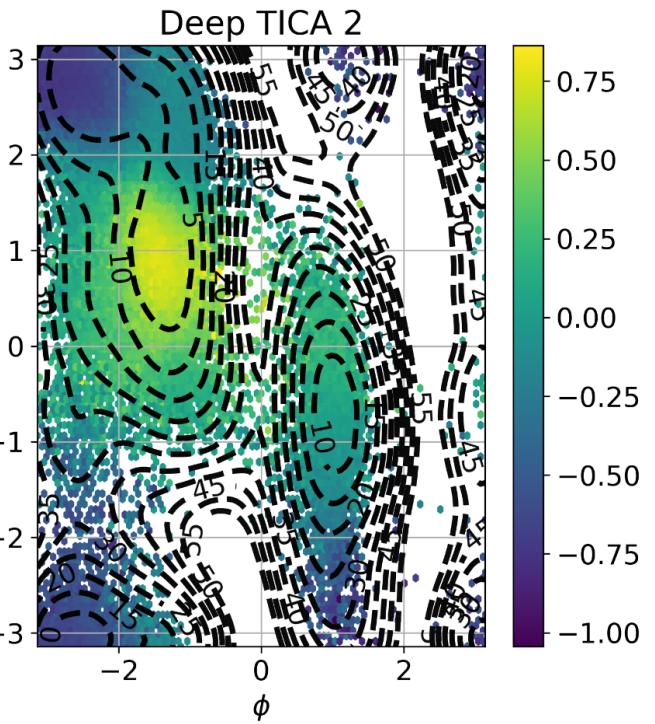
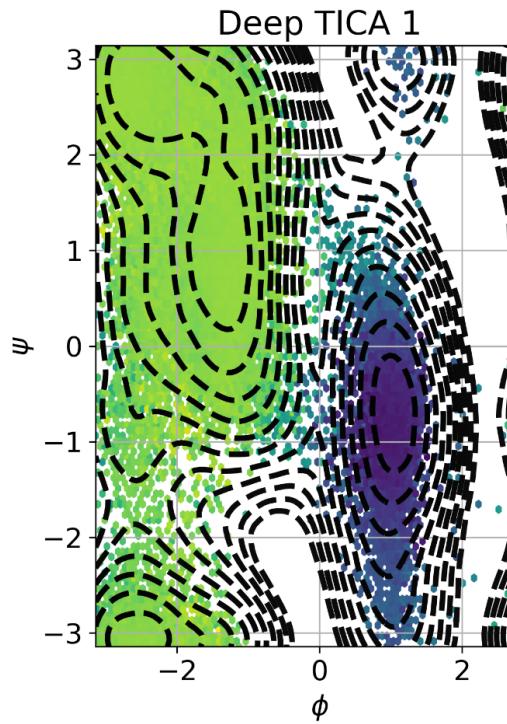
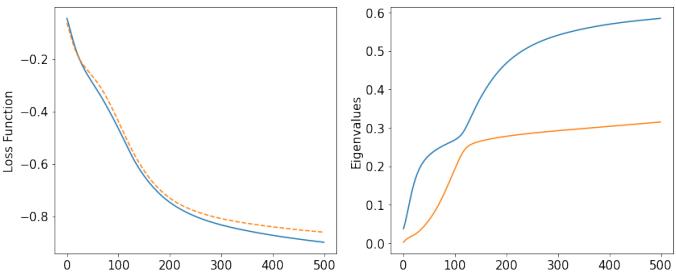
Deep TICA 1



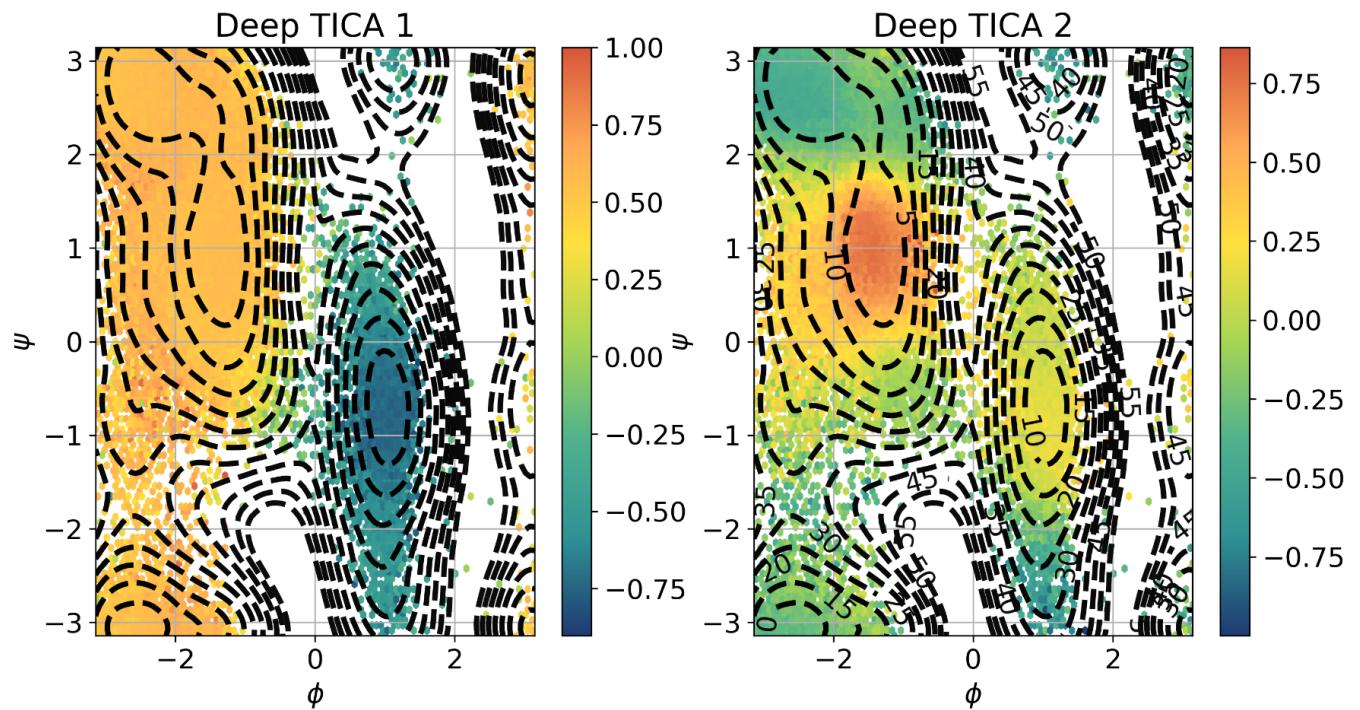
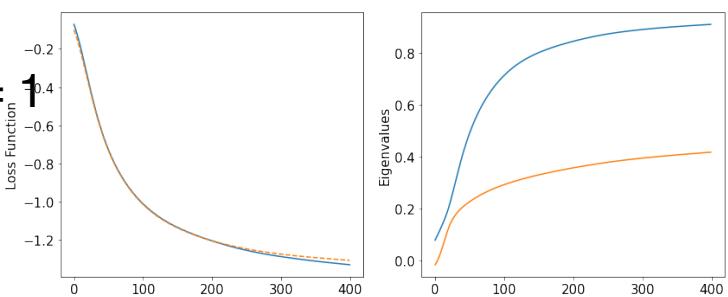
Deep TICA 2



Deep TICA Analysis, rescaled as option (2) ,lag time = 1



Deep TICA Analysis, rescaled as option (3) ,lag time = 1



Perspectives

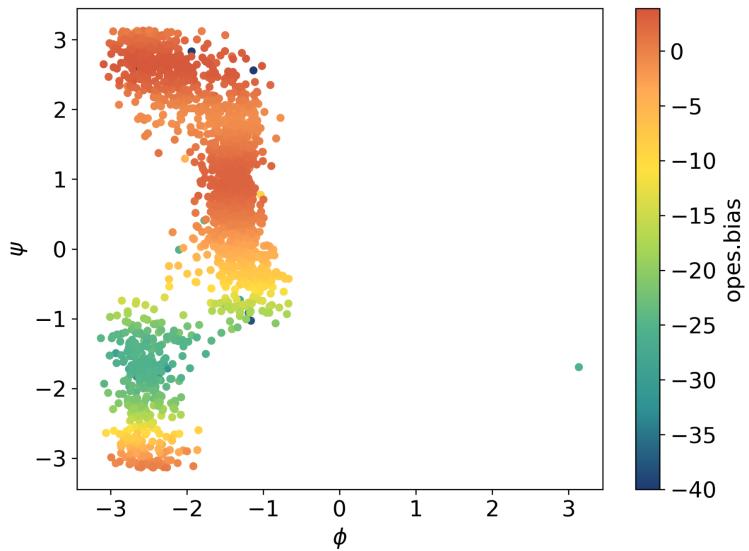
One may think that it is not important to study a biased diffusive dynamics, because the most common situation is dealing with a simulation that either exhibits few transitions between metastable states (slow modes detected both rescaling and not rescaling the time) or even worse that does not exhibit transitions at all.

The latter scenario is the most common for short unbiased trajectories at low temperatures, where the system remains stuck in a metastable state due to kinetic and entropic bottlenecks.

For this reason I analyze the first and the second part of the biased simulation along Psi. Within the first steps, for almost 1.8 ns, the system remains stuck in the first basin (A ,the deepest one). Then from almost 2ns to 35ns stays fixed in the second state (B).

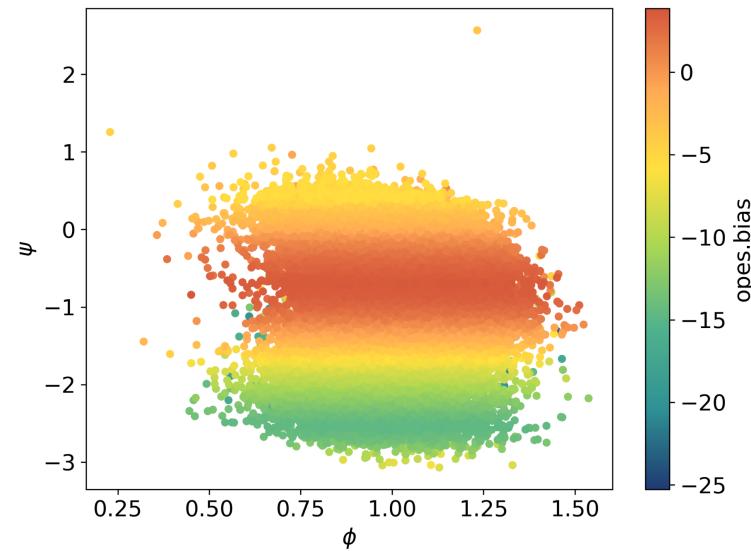
One may expect that after performing the Deep-TICA and TICA Analysis the slowest mode must be that one orthogonal to Psi, because in this direction the dynamics has been already accelerated.

First part, 0-1.8ns



1

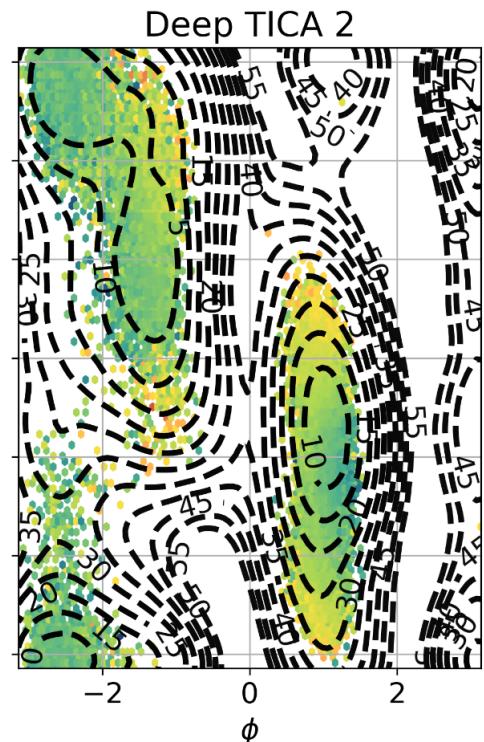
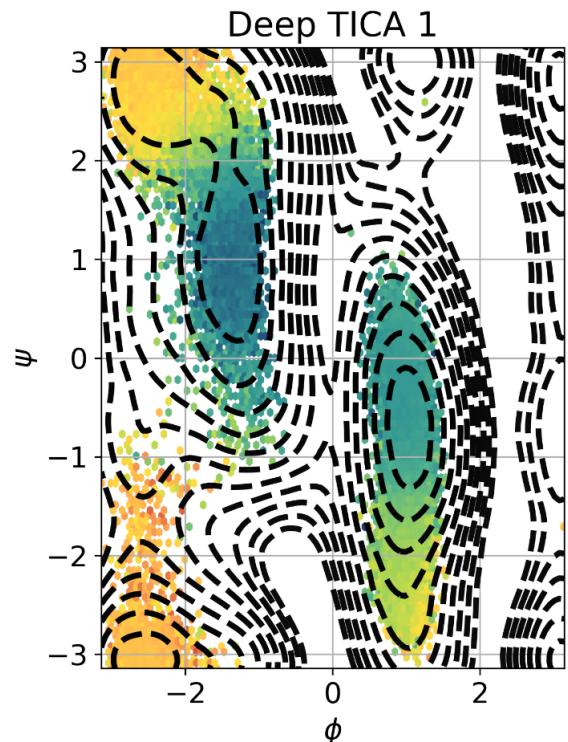
Second part, 2ns-35ns



2

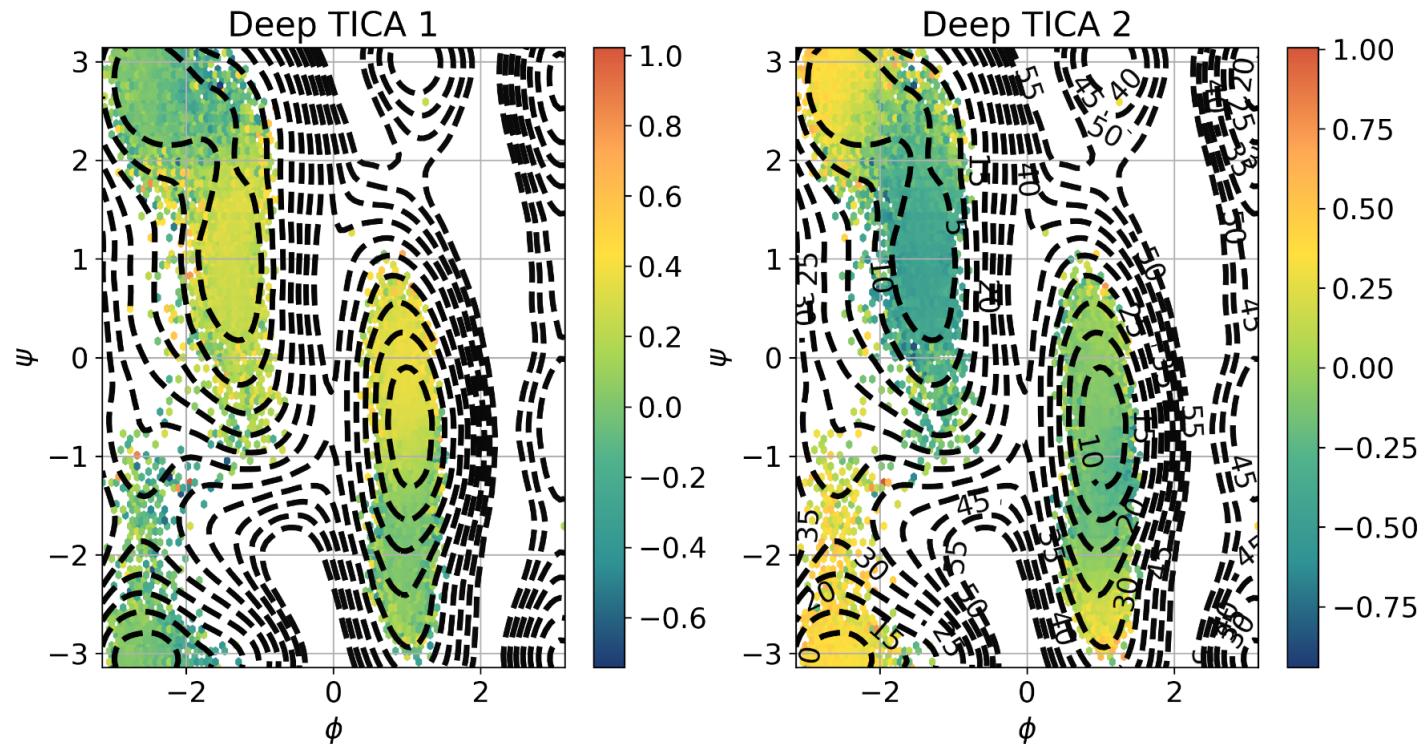
Deep TICA Analysis, without using the rescaled time,
lag time = 1

1



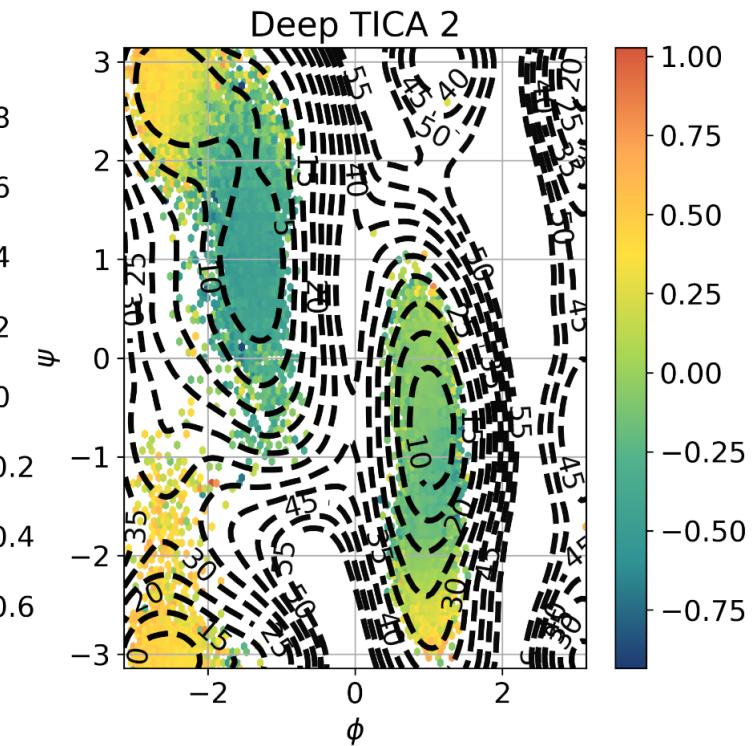
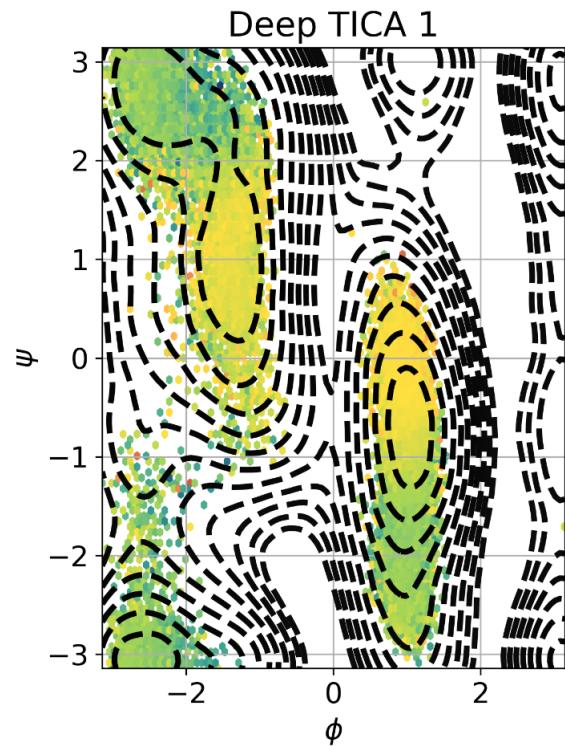
Deep TICA Analysis, rescaled as option (1) ,lag time = 2

1



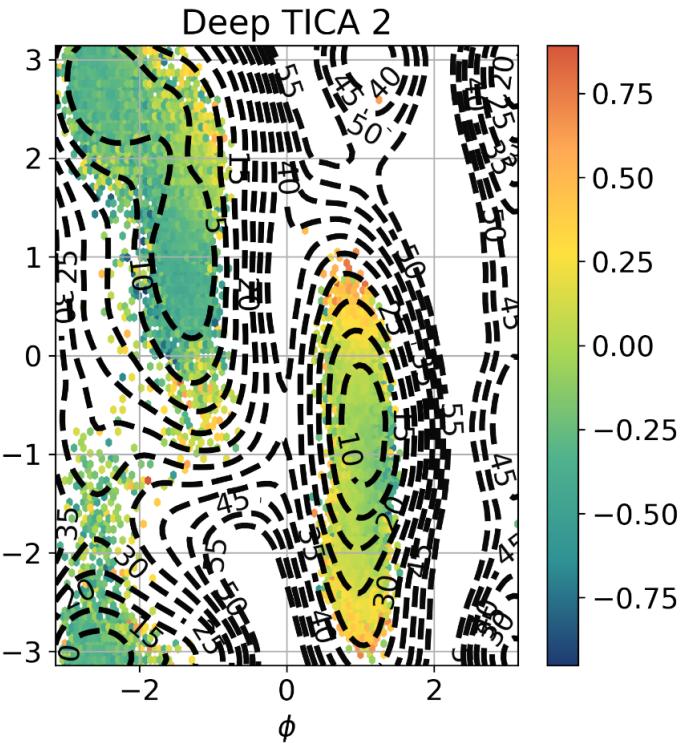
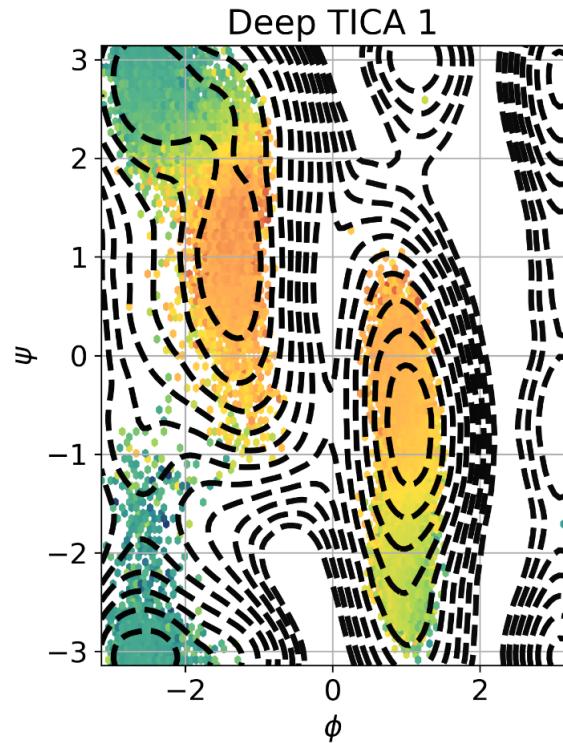
Deep TICA Analysis, rescaled as option (2) ,lag time =
0.5

1



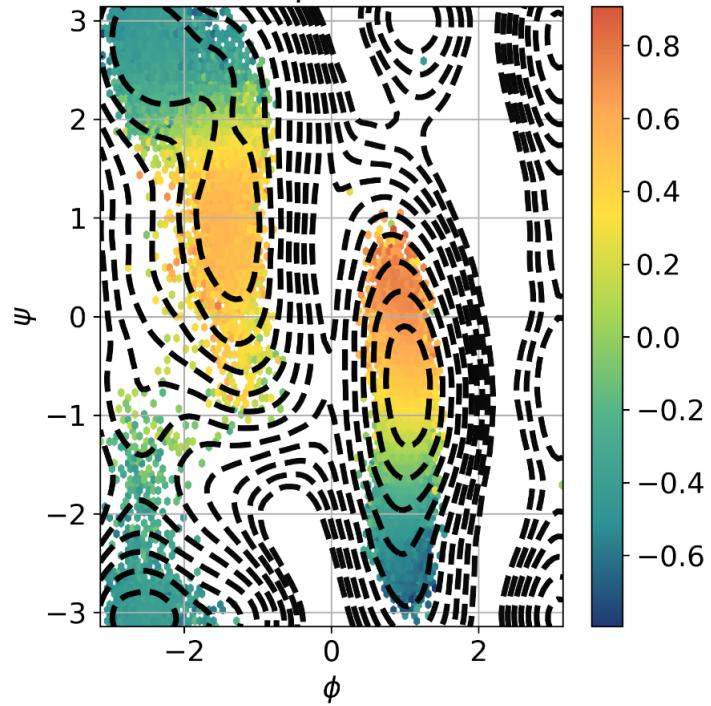
Deep TICA Analysis, rescaled as option (3) ,lag time = 1

1

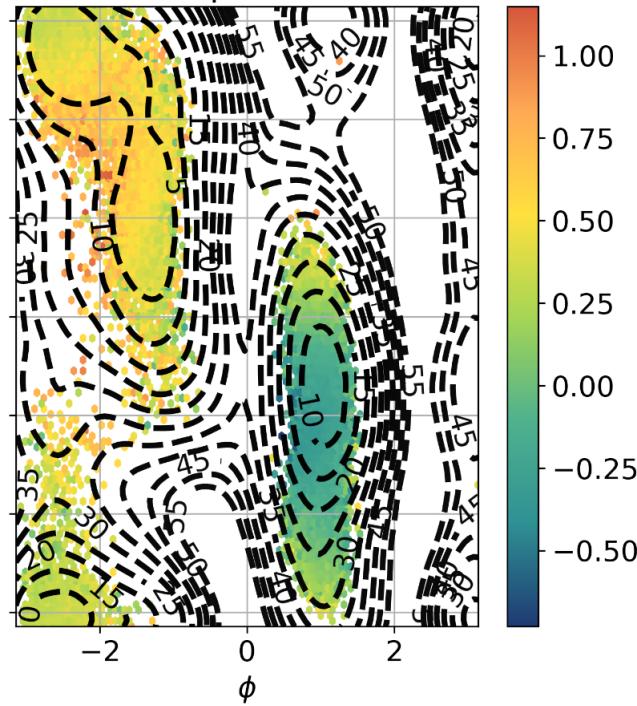


Deep TICA Analysis, without using the rescaled time, lag time = 2

Deep TICA 1

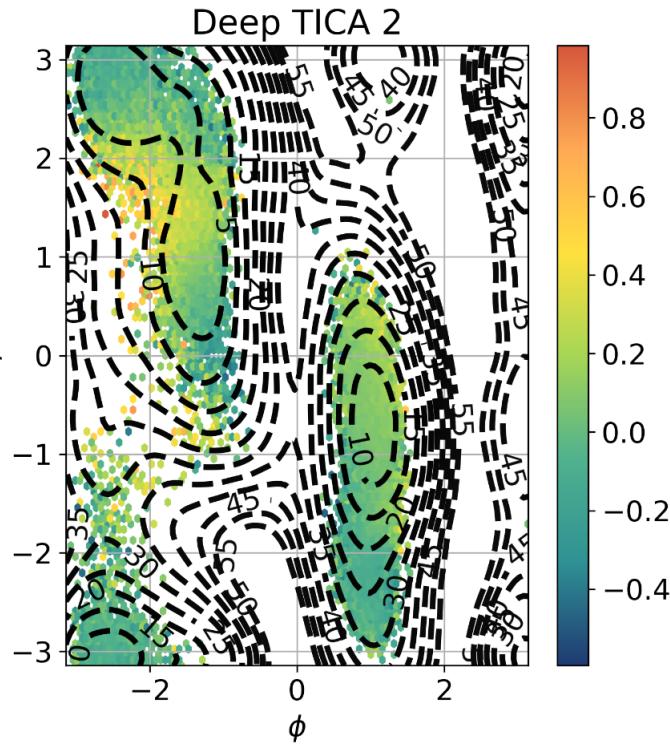
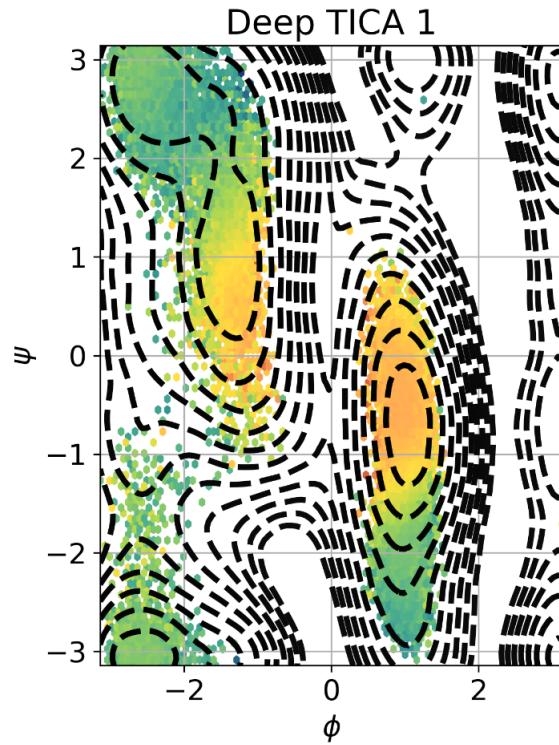


Deep TICA 2



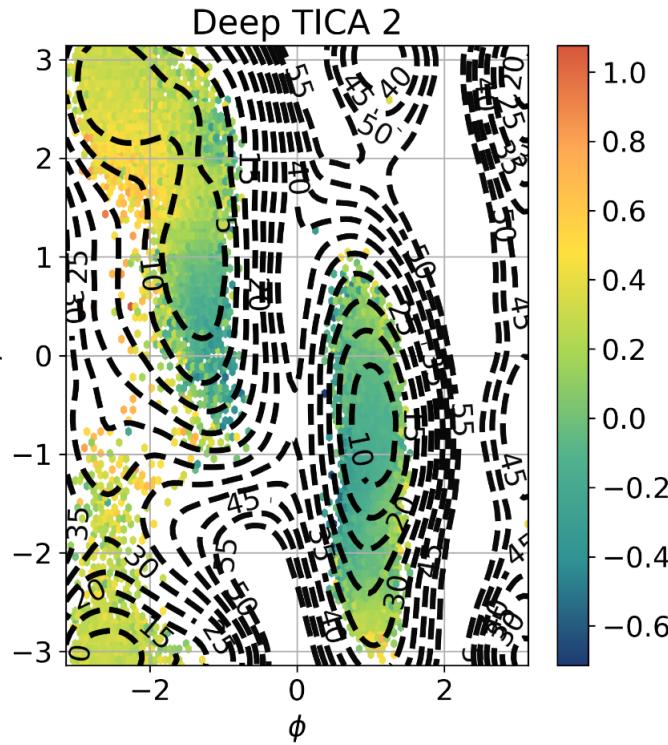
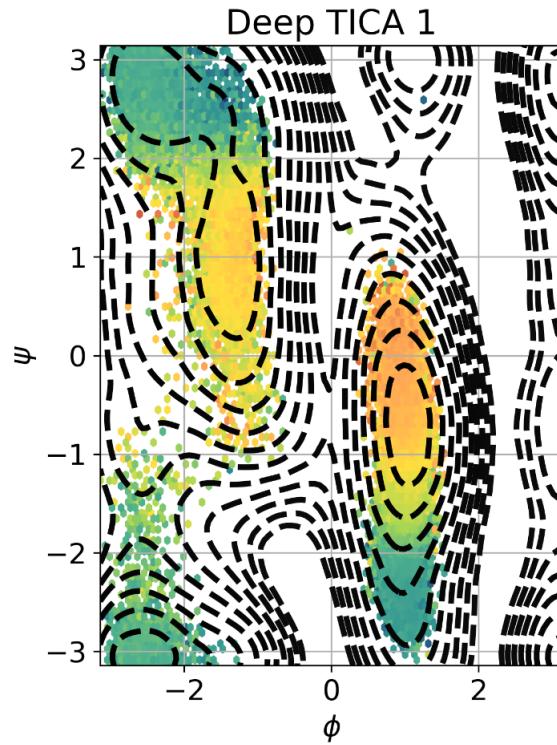
Deep TICA Analysis, rescaled as option (1) ,lag time = 3

2



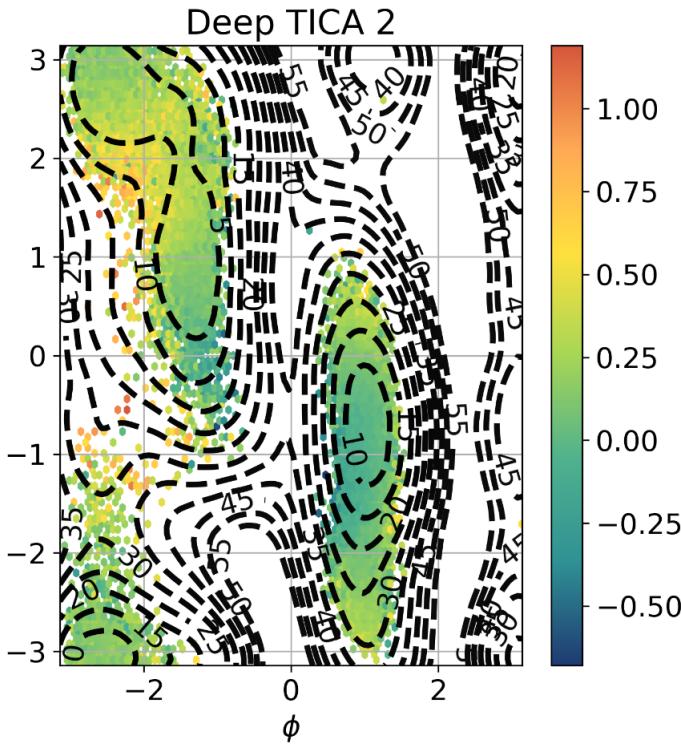
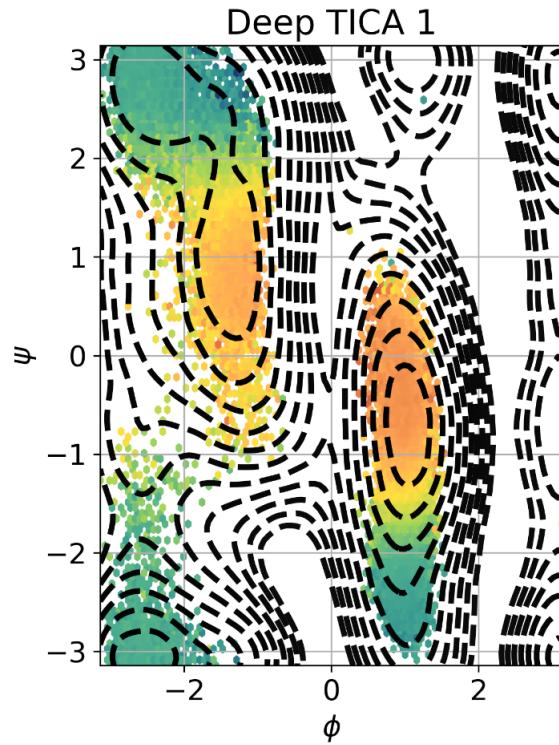
Deep TICA Analysis, rescaled as option (2) ,lag time = 1.5

2



Deep TICA Analysis, rescaled as option (3) ,lag time = 1.5

2



Comments:

None of them is able to find the Phi as slow mode. But this is coherent with the previous results.

If the target distribution is well tempered and there exists a slow mode (which actually is a fast mode compared to the equilibrium distribution, but inside a basin can be considered as slow) according to which the data are distributed, then this slow mode is slower than that mode/motion relative to the fluctuations that can lead the system to escape from the basin.

I recall that the training and the TICA analysis is really sensitive to the choice of the lag time hyperparameter. I always choose a lag time that can give the better comparison between validation loss function and training loss function. The rescaling option (1) and (2) are more sensitive to this choice.

Later I will show a strategy to overcome the problem of sensitivity for the choice of the lag time value.

My strategy:

My idea is to apply in a iterative matter the Deep-TICA Analysis. So that at each iteration the algorithm is able to find a suitable set of CVs that can be used in the following enhanced sampling simulation.

These CVs are (in principle) able to accelerate the local (inside a basin) slow motions of the system under study. If more basins are present that the first slow mode will describe the transitions between these two states. If no sub basins are present than the first slow mode simply follows the directions relative to the slowest fluctuations. For instance the direction along with the barrier is less steep.

This direction is not in principle the correct one, able to let you escape from the local minima, but in the next simulation will bring the system outside that particular direction and in a more natural manner it will reach a new metastable state.

Then in order to improve the convergence and stability of the training I'll train in batches the network, where each batch is a set of data obtained from the different previous simulations.

