**Identifying optimal precursors for two types of El Nino events based on CNOP with GFDL CM**

**Abstract：**

Conditional nonlinear optimal perturbation (CNOP) has been used to study the optimal precursors (OPR) for two types of El Nino events for years, and the OPR for EP-El Nino has been found with many climate models. However, the OPR for CP-El Nino event is not satisfactory, the reason lies on that many climate models do not have the ability to simulate CP-El Nino event, while GFDL CM has strong capabilities for simulating two types of El Nino events. So, identifying optimal precursors for two types of El Nino events based on CNOP with GFDL CM will be a significant attempt. GFDL CM is a global ocean-atmosphere coupled model with a high resolution and no adjoint model. Intelligent algorithm is an adjoint-free method and has been used in many climate models to solve CNOP successfully reaching thousands of orders of magnitude. The dimensions of GFDL CM is up to hundred thousand for scenarios of simulating two types of El Nino events, so efficiency must be considered. In this paper, based on particle swarm optimization (PSO) algorithm, we propose a dynamic feature space particle swarm optimization (DFSPSO) algorithm aiming at accelerating solving CNOP of GFDL CM. Then we identify optimal precursors for EP-El Nino and CP-El Nino event. We evaluate the effectiveness of DFSPSO from the CNOP value, convergence speed, stability, and computation time. And we evaluate the effectiveness of OPRs for EP-El Nino and CP-El Nino event by superimposing the corresponding OPRs to initial state from Nino index and SSTA pattern. The experimental results show that DFSPSO can converge quickly and stably, obtain the OPRs of EP-El Nino and CP-El Nino event effectively.

***Key word: optimal precursor; EP-El Nino; CP-El Nino; CNOP; GFDL CM; DFSPSO***

1. **Introduction**

El Nino-Southern Oscillation (ENSO) events are coupled ocean-atmosphere phenomena in the tropical Pacific and have received much attention for both their climatic and economic impacts (Rasmusson and Carpenter,1982; Barber and Chavez,1983; Cane,1983; Rasmusson and Wallace,1983). One significance aspect of ENSO predictability research is to address the nature of the initial perturbations that is most likely to develop into ENSO events. This kind of initial perturbation, in some conditions, is regarded as the optimal precursor (OPR) of ENSO event. Finding the OPR that can develop into ENSO event can help us understand the mechanism of ENSO development, the process of ENSO, what’s more, improve the prediction skills. Conditional nonlinear optimal perturbation (CNOP) has been applied to study the OPR of ENSO events for years, which is characterized by maximum nonlinear growth of the initial perturbation in a given condition.

Based on the interannual variation and spatial distributions of the sea surface temperature (SST), More and more evidences have shown that there are two different types of El Nino events in the tropical Pacific Ocean: the eastern-Pacific (EP) El Nino and the central-Pacific (CP) El Nino. Most numerical climate model have a good ability to simulate EP El Nino, but the ability to simulate CP El Nino is not strong. So, there are many researches on the OPR of EP El Nino, …

There are also some researches on the OPR of CP Cl Nino. Xu (2014) used different cost functions to simulate OPRs for two types of El Nino by CNOP in Zebiak-Cane (ZC) model, the result shows that the OPRs can develop into EP-El Nino event and “mixed El Nino event” (Kug et al. 2009) except La Nina event. Duan et al. (2014) used optimal forcing vector approach to simulate EP-El Nino and CP-El Nino events as well as in ZC model, which can reduce the effects of model errors.

Geophysical Fluid Dynamics Laboratory Climate Model (GFDL CM) is a complex global ocean-atmosphere coupled model with subsurface processes that can be used to study the predictability of ENSO events (Kug et al., 2009, 2011). It includes four modules: Atmosphere, Ocean, Sea Ice and Land. The main two modules related to ENSO are the first two modules. The atmospheric module is AM2.1 with a resolution of 144 grid points in longitude, 90 grid points in latitude, and 24 layers in the vertical direction. The ocean module is MOM4p1, with a resolution of 360 grids of longitude and 200 grids of latitude, and 50 layers of vertical layering, of which the resolution of the upper layer of 225 meters is 10 meters per layer. Atmospheric and ocean modules exchange component flux every two hours.

It has been proven GFDL CM has the ability to give a good simulation for two types of El Nino events(文献). In this paper, GFDL CM is chosen to identify the OPRs of two types of El Nino events. In other words, to identify the OPRs of two types of El Nino events is to solve the CNOP of GFDL CM. identifying optimal precursors for two types of El Nino events based on CNOP with GFDL CM will be a significant attempt.

The need of integrations of adjoint model during solving CNOP, however, is limited in complicated coupled models that do not have an adjoint model such as GFDL CM. GFDL CM has a high dimensional resolution but no adjoint model. To avoid this restriction, adjoint-free method must be used. Intelligent algorithm is one of adjoint-free methods, which are based on feature extraction and heuristic search and can get similar results compared to the adjoint method. They have been used to solve CNOP successfully in ZC、MM5、WRF、ROMS、CESM (文献). However, the dimensions of related parameters in GFDL CM is up to hundred thousand when solving CNOP when simulating ENSO event, in the supercomputer environment, it takes four hours to integrate one year, and a total of 350 years is required, efficiency should be considered. The previously proposed intelligent algorithm based on feature extraction has very low computational efficiency even if it is reduced to a lower feature space. And if the feature space is fixed when heuristic search, the feature space will only contain the characteristics of initial samples but not catch the characteristics of the global optimal and the convergence of intelligent algorithm will be very slow, which will cause the intelligent algorithm to slow down further.

So, in this paper, based on particle swarm optimization (PSO) algorithm, we propose a dynamic feature space particle swarm optimization (DFSPSO) algorithm aiming at accelerating solving CNOP of GFDL CM, which means the feature space is dynamically changed between iteration, the feature space will seize the characteristics of the global optimal gradually. DFSPSO can accelerate the convergence and achieve better results with fewer iterations. And the skill of space-time conversion and matrix perturbation will be used between iteration to accelerate the iteration of feature space.

The initial perturbation will be superimposed into sea surface temperature and sea subsurface temperature. According to the different characteristics of the two types of El Nino events, the initial perturbation will be superimposed in different ranges when using DFSPSO solve CNOP in GFDL CM. Then we identify optimal precursors for EP-El Nino and CP-El Nino event. We evaluate the effectiveness of OPRs for EP-El Nino and CP-El Nino event by superimposing the corresponding OPRs to initial state from Nino index and SSTA pattern. From these two aspects, verify whether the obtained OPRs can develop into corresponding two types El Nino events and features. At the same time, we evaluate the effectiveness of DFSPSO from the CNOP value, convergence speed, stability, and computation time by comparing it with PSO algorithm. The experimental results show that DFSPSO can converge quickly and stably, obtain the OPRs of EP-El Nino and CP-El Nino event effectively.

The rest of the paper is organized as follows: Section 2 introduces the related work. In Section 3, the dynamic feature space particle swarm optimization method is presented. How to solve CNOP in GFDL CM will be presented in Section 4. Experiments are presented in Section 5. This paper ends with the conclusion and future work in Section 6.

1. **Theory and method**

**2.1 CNOPs**

Mu et al. proposed CNOP to study the predictability for weather and climate. Consider a nonlinear partial differential equation:

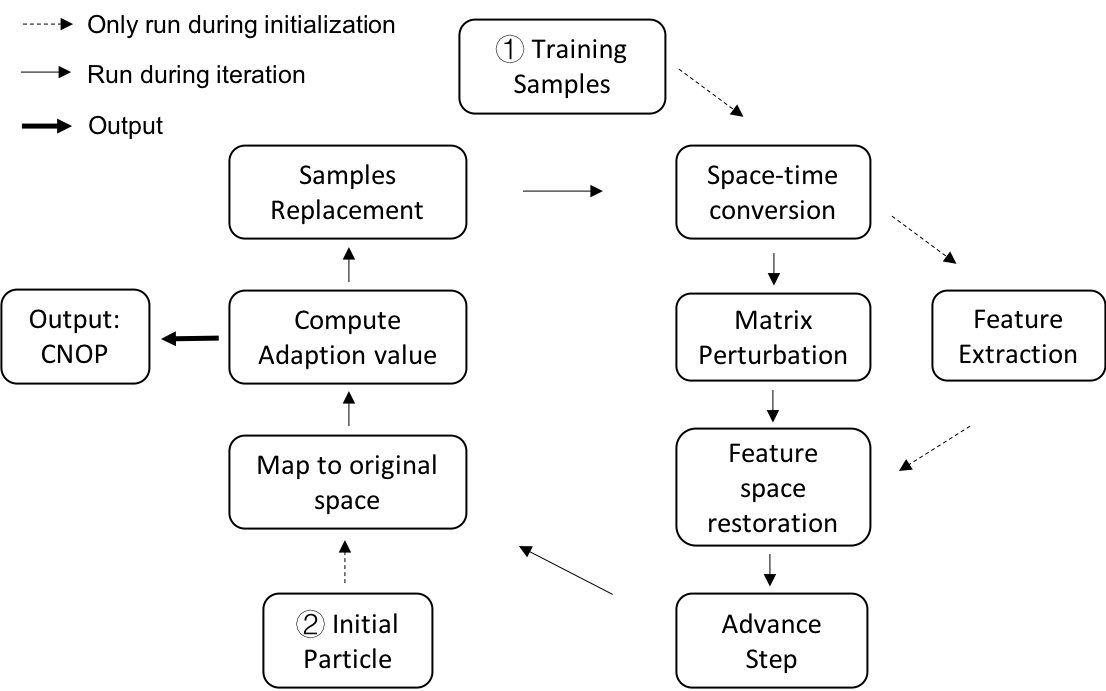
Where is the state vector, is a nonlinear operator, is the time. For the given initial field , the solution to Eq. (\*\*\*) is given by

where denotes the propagator of the nonlinear from time to time prediction time , GFDL CM will be used for nonlinear integration with given initial state.

* 1. **Dynamic feature space particle swarm optimization (DFSPSO)**

In this paper, based on PSO, we propose the DFSPSO to solve CNOPs for identifying OPRs of two types of El Nino events. The core of this algorithm is to replace sample and to change feature space dynamically during iteration. PSO is a classic population-based heuristic optimization algorithm developed by Kennedy and Eberhart (1995) and inspired by the social behaviors of bird flocking or fish schooling. The technique has been successfully and effectively applied to solve CNOPs in the ZC model for studying ENSO predictions (Mu et al., 2015a).

At the beginning, we have made the attempt to adopt PSO to solve CNOPs in GFDL CM, although the results exhibit slow convergence and insufficient search. Hence, we will overcome these shortcomings by dynamically changing the feature space during iterations. Fig. (\*\*\*) shows the framework of DFSPSO.



In Fig. (\*\*\*), the dotted arrows run only during initialization, the solid arrows run during iterations and thick solid arrows run as output When meeting the iteration termination condition. The most important two processes of DFSPSO is: 1) feature space extraction of training samples and 2) iterative optimization. In 1), because of the complexity of ocean-atmosphere grid data in GFDL CM, which indicates that the number of dimension is much larger than the number of samples, if we use the traditional method of feature decomposition, it will take a very long time, so a space-time conversion in PCA will be adopted when calculating feature space of samples. In 2), dynamically update feature space to speed up optimization where some accelerate approximate calculation methods are also used. Some adaptive dynamic advance step methods have also been added to the original PSO algorithm: using large steps to increase the search space at the beginning of the iteration for extend search domain, using small steps at the end of the iteration for accelerate convergence where searching near the optimal value. Some skills in DFSPSO are described as follows:

* + 1. ***Feature extraction***

The most important indicator distinguishing two types of El-Nino events is the sea temperature including sea surface and sea subsurface. In GFDL CM, the ocean module is divided into 50 levels and the resolution of each level is . It is necessary to reduce the dimension at such a high resolution. A lot of studies have shown that any space can be projected onto a low-dimensional space and be represented approximately by the attractors (Osborne A R, Pastorello A 1993; Foias C, Temam R 2010). According to this conclusion, PCA is an effective way to obtain the attractors of chosen samples.

The preprocessing of feature extraction is centralization of samples, that is,

After centralization, PCA will be used for dimension reduction. Supposed the sample Matrix is , where represents the number of grids and represents the number of samples. The principal components (PCs) can be acquired with eigenvalue Decomposition:

where represents the eigenvalues of with a descending order which is a diagonal matrix, and is composed by the eigenvectors referred by eigenvalues . In common sense, the number of grids is much large than the number of samples (), will be a very high dimensional matrix where the calculation process takes lots of time and effort. So, Space-time conversion will be used instead:

where . And the non-zero eigenvalues in and are the same. The relationship between and is as follows:

If the samples are independent each other or orthogonal, the number of eigenvalues for is , the number of eigenvalues for is , that is, only the first eigenvalues can be obtained. However, the actual application has little effect on the results, because usually only the first few most important eigenvalues will be focused on.

* + 1. ***Dynamic feature space and iteration***

Usually the feature space is not changed in the algorithm. In the end of the iteration, however, the characteristics of optimal value cannot be well described. That is, the meaning of samples is usually only a quantitative accumulation. Most of the samples only have general characteristics but do not have the features we are looking for. The feature space calculated by such samples cannot grasp the characteristics of the optimal value at the end of the iteration, because most of the particles have evolved into particles with certain special features at the end of the iteration, and it is obviously not suitable to continue to use the original sample feature space.

The use of dynamic feature space in the iterative process is conducive to seizing the features searching for, and it can speed up the convergence and make the particle swarms reach the optimal value in advance. The dynamic feature space consists of sample replacement and re-calculate feature space. For speed up the calculation, a method of matrix perturbation is used.

1. ***Sample replacement***

In each iteration in PSO algorithm, there are always some particles who have the larger adaption value according to adaption function, and they are regarded as better particles. Moreover, these better particles are “closer” to global optimum (or local optimum), so they are supposed to seize more features we want. Replace them in samples and re-calculate the feature space, the feature space will have characteristics closer to the optimal value. We choose to random replacement of the original samples.

1. ***Matrix perturbation***

When the samples have changed, the feature space and eigenvalue decomposition should be re-calculated, which is a consumption of time and memory. So, the method of matrix perturbation is used, which is based on the difference in samples between iterations is small enough, that is,

where represents the samples of next iteration and represents the samples of last iteration. And if only a few particles replace into samples, will a very sparse matrix, and non-zero values will be small. suppose , represents the number of iterations, so the Eq. (\*\*\*) can be written as,

According to Rayleigh quotient (Horn, R. A. and C. A. Johnson. 1985; Parlet B. N 1998),

where is left eigenvector corresponding to eigenvalues, is right eigenvector corresponding to eigenvalues. Because is a hermite matrix, the left eigenvector is equal to right eigenvectors, and if has centralization process. So, expressed in the form of a matrix, Eq. (\*\*\*) is

According to Matrix perturbation and Rayleigh method (Lloyd N 1997), the above formula needs some corrections,

After the above formula corrections, the and can be approximated as the eigenvalue and eigenvector of the samples.

1. ***Iteration***

The dynamic feature space and dynamic step are added into the traditional PSO algorithm (J. Kennedy, R. Eberhart 1995), the advance iteration formula is as follows:

where represents the speed of th particle in th iteration, and represents the location of th particle in th iteration, represents the best location of th particle before th iteration, represents the best location of all particles before th iteration. is a random number in . Parameter , and will change dynamically as following:

where represents the max step of iterations, represents the number of current iterations. In this way, and are monotonically decreasing functions, and is a monotonically increasing function. Using large steps to increase the search space at the beginning of the iteration for extend search scope, using small steps at the end of the iteration for accelerate convergence where searching near the optimal value. The DFSPSO algorithm is shown as following:

Algorithm DFSPSO

|  |
| --- |
| **Input:** a training set |
| **Initializations:** |
| 1: Set the parameters , , , |
| 2: Calculate the constraint |
| **Feature Extraction:** |
| 3: Perform the centralization of |
| 4: Make the eigen-decomposition of matrix |
| 5: Calculate the eigen value of |
| **DFSPSO:** |
| 6: Randomly generate an initial swarm |
| 7: project swarm into original space |
| 8: **while** the termination condition is not satisfied **do** |
| 9: Calculate the adaption values |
| 10: Update global and local optimal positions |
| 11: **if** the current generation is satisfied **then** |
| 12: randomly generate the perturbations |
| 13: update the position of each particle |
| 14: **else** update the position of each particle |
| 15: **end if** |
| 16: replace better sample into original sample |
| 17: re-calculate the eigen value and eigen matrix by matrix perturbation |
| 18: project the particle back to original space |
| 19: **end while** |
| **Output:** the optimal swarm |

1. **Solving CNOP of GFDL CM with DFSPSO**

The most important thing to apply DFSPSO to solve CNOP is to associate its particles with the physical quantities in GFDL CM. We use perturbations on sea temperature including sea surface temperature and sea subsurface temperature to identify two types of El Nino event, so the particle in DFSPSO is represent the grid data on sea temperature in GFDL CM. Suppose a particle in low dimension, according to the formula for projecting particles back to the original space

where represents the perturbation on sea temperature in GFDL CM, can be obtained from Eq. (\*\*\*\*) which is feature space of origin samples. And the perturbation which makes the target function get the maximum with the condition of , as follows:

where is the nonlinear evolution of the initial perturbation , is the constraint of perturbation, and denotes the norm. Every particle needs to satisfy the boundary constraint. The constraint of sea temperature perturbation after swarm projecting back to original space is defined as follows:

where represents the location of grid in GFDL CM, represents the latitude of gird, represents the sea temperature perturbation of gird after the swarm projecting back to original space, represents the standard deviation of sea temperature perturbations of gird in samples. presents the constraint which can be determined according to the value of each sample calculated by this formula (Sorting in descending order, taking value of 10% of the ranking).

During iterations, due to the randomness of the parameters, the particles may evolve into a larger value, if projecting them to the space of GFDL CM, they may present a larger sea temperature which does not meet the physical meaning. Once the sea temperature perturbation after swarm projecting back to original space exceeds the constraint, the following formula can remap particle back into constraint:

where represents the swarm updated, represents the value of Eq. (\*\*\*) for this swarm.



Fig. (\*\*) shown the process of solving CNOP with DFSPSO algorithm. The process of solving CNOPs with DFSPSO algorithm is described as follows:

1. *Use PCA to generate the feature space.* All of the processes are based on the dimension reduction within the PCA, and space-time conversion will be used when Feature decomposition.
2. *Randomly generate an initial swarm with N particles.* Every particle needs to satisfy the boundary constraint according to Eq. (\*\*\*). Once the particle is out of bound, it must be scaled to within the bound according to Eq. (\*\*\*).
3. *Project the particle back into original space and Calculate the adaption value of object function.* i.e.,  in Eq. (\*\*\*)
4. *Update particles by PSO and randomly replace the samples. Use matrix perturbation re-calculate the feature space.* The iterative rules of the particles follow Eq. (\*\*\*). And described in Eq. (\*\*\*), the related parameters are updated dynamically during iteration.
5. *Judge whether the termination condition is satisfied.* So, terminate the iteration. Otherwise, go to step 3.

After many experiments, the parameters of the ACPW algorithm can be set, as shown in Table (\*\*\*).

|  |  |  |
| --- | --- | --- |
| Name | Meaning | Value |
|  | Number of principle components |  |
|  | Number of particles |  |
|  | Constraint |  |
|  | Inertia coefficient |  |
|  | Self-awareness to track the historically optimal position |  |
|  | Social awareness of the particle swarm to track the globally optimal position |  |
|  | Number of samples replaced during iteration |  |
|  | The number of iterations |  |

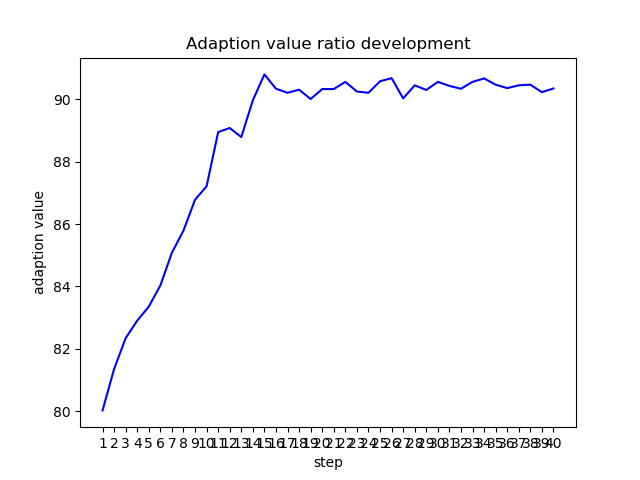
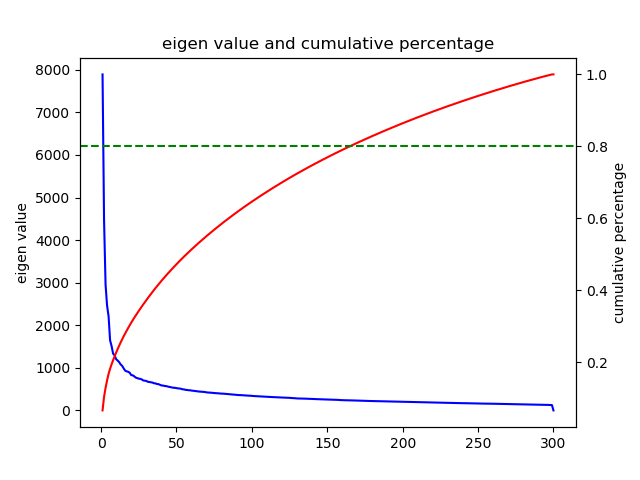
Although DFSPSO is almost the same number of parameters as PSO, these parameters can lead to better results with almost no adjustment. Perhaps a combination of parameters in PSO can achieve better results, but finding this combination is time consuming. The above combination of parameters is enough to meet the experimental requirements.

1. **Experiment results analysis**
   1. ***Environment and parameter settings***

GFDL CM is a high-resolution ocean and atmosphere model which needs at least 45 cores to run, and DFSPSO is a kind of parallel particle swarm algorithm, so supercomputer (Tianhe2) is the best choice for this experiment. All the experiments are run on Tianhe2 system. All the codes are written in the Java language and version is JDK8. GFDL CM is compiled by Ifort11. Due to limited resources, DFSPSO is set to run 10 particles in parallel at a time. The samples are generated by GFDL CM itself which is the natural integration of the model for 350 years and discard its previous 50 years of unstable data which means the number of samples is 300. For different CNOP for two types of El-Nino, and are varied, the following chapter will give an introduction. As for solving OPR for two types of El-Nino, the “Spring prediction barrier” is an unavoidable question, so according to Duan W, Wei C (2013), the January is chosen as start month, and December is chosen as predict month which can eliminate its impact as much as possible.

* 1. ***Comparison with PSO algorithm and result analysis***

In traditional PSO algorithm, the feature space is fixed during iteration and particle iteration in high dimensional space is time consuming, when using PSO algorithm, the number PC and performance are supposed to well chosen by dimensionality reduction (参考文献). An experiment has been set with PSO algorithm with a swarm size of 60 and the number of iterations is up to 40. The dimension of feature space is chosen at 165, which means the cumulative percentage of eigen values is up to 80% as Fig. (\*\*\*).



As Fig. (\*\*\*), the beginning number of eigen values is also chosen at 165 in DFSPSO which means the percentage of eigen values is up to 80% too. However, as the number of iterations increases, the feature space continues to iterate in DFSPSO as Fig. (\*\*\*), although the number of eigen values is fixed, the percentage still increases. It means the proportion of eigenvalues at the beginning of the iteration is selected instantly, and it will increase with sample replacement and feature space iteration. The reason why the proportion of eigenvalues tends to be constant at the end of iteration may be that 1. Sample replacement has randomness, and all samples cannot be replaced once; 2. Without selecting all eigenvalues, the selected dimension cannot occupy the full proportion.

For the case where 80% of the eigen values are selected, the PSO algorithm was not able to converge easily and the algorithm is at least 5 times as slow as DPSPSO. So even if a good solution could be found when using a larger swarm size and more iterations, DFSPSO would still outperform PSO in terms of time consuming of iteration. An explanation for the inefficiency might be that the feature space cannot catch the characteristics of ENSO. Even if the optimal parameter settings are found, it will take a long time to do a lot of experiments and may take some research based on the characteristics of ENSO. Aleid (XXXX) have pointed out that while there might exist parameter settings for some algorithm in specific problem which is itself an optimization problem, maybe it will solve your specific problem successfully if they are tuned perfectly, but they are hard to find and even the problem cannot be solved when the parameters are not tuned well. So, in DFSPSO, we can try to avoid the choice and tuning of parameters and spend a small amount of time to get better results.

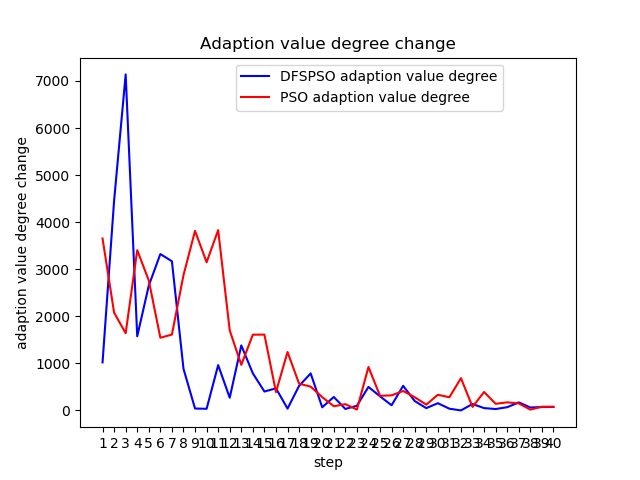
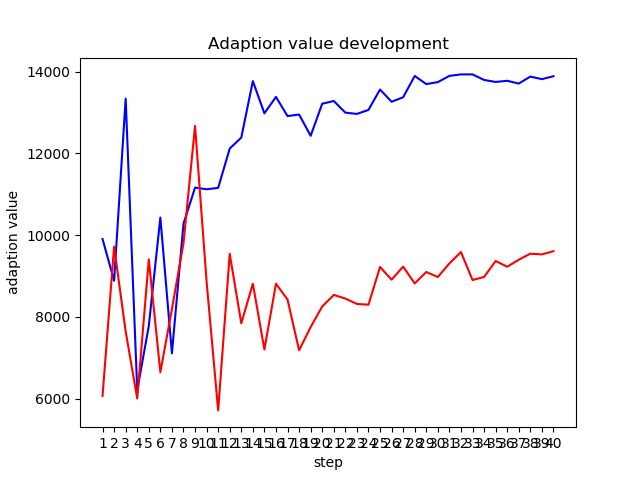
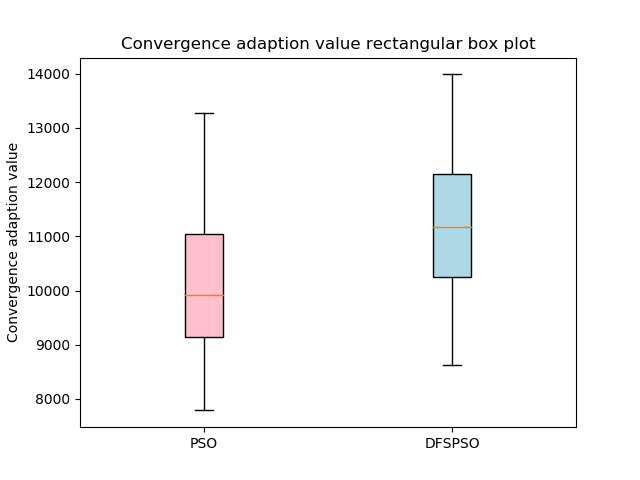
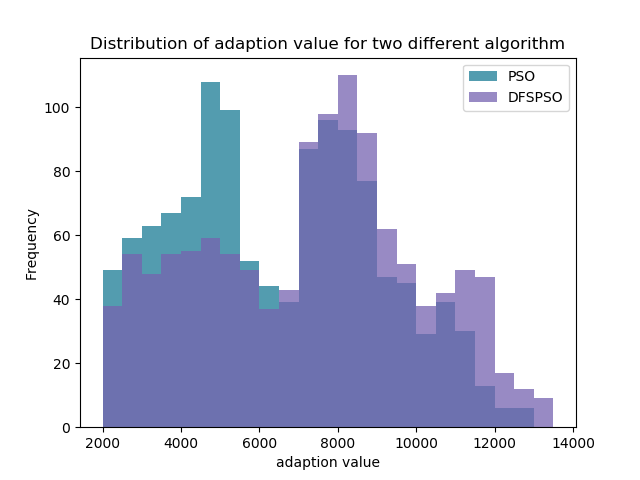


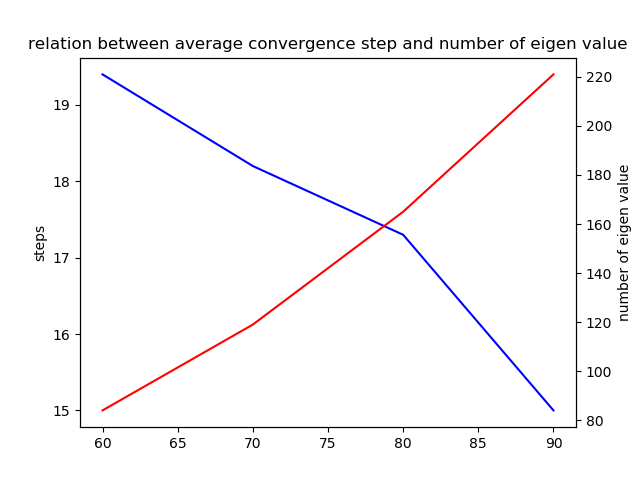
Fig. (\*\*\*) shows the average adaption value development of PSO and DFSPSO in the experiment, which indicates that: in DFSPSO, at the beginning of the iteration, the adaption value rapidly developed and reached the peak; during the middle of the iteration, the particle swarm increased the search range and searches for other fitness areas in the possible range which may cause a decline of adaption value; At the end of the iteration, the particle swarm quickly converge to the global CNOP. In PSO, the adaption value fluctuates violently until the middle of the iteration does not converge, at the end of the iteration, the adaption value is not as large as DFSPSO, and it is still in a slow upward trend. In Fig. (\*\*\*), the degree of change is calculated from the subtraction of two objective values in PSO and DFSPSO. For example, the objective value of the second iteration minus the first objective value is the first degree of change. Obviously, the DFSPSO algorithm is continuously changing in the beginning of iteration and stable at the end of the iteration and local search near the global optimal. And PSO algorithm cannot converge, although the change of adaption value has been going on, but from the Fig. (\*\*\*), it is still in the state of optimization, the change of the adaption value is small, and the speed of optimization is slow. Above all, we can conclude that the DFSPSO algorithm has better performance than the PSO, because we overcome the shortcomings of the PSO search limitations and add the skill of accelerating iteration.

In Fig. (\*\*\*), because CP El Nino experiment has two obvious optimal values, we perform PSO and DFSPSO in CP El Nino experiment for 10 times and separately perform the results of PSO and DFSPSO algorithms to make the distribution of the adaption values into a histogram. In 10 experiments, the PSO has the narrow scope, from 2000 to 13500. The DFSPSO algorithm has a larger value spans that are wider than the PSO, but the distribution of objective values of the DFSPSO are higher. And the value scope is reasonable according to the characteristics of these two algorithms. PSO has weaker divergence ability than DFSPSO, which means the peak of distribution of PSO is located in [5000, 6000] and [7000, 9000] but the peak of distribution of DFSPSO is located in [7500, 9000] and [10500, 12000]. According to the CP El Nino experiments, we know that the adaption located in [7500, 9000] is a CP El Nino event which is local optimal, and located in [10500, 12000] is a La Nina event which is global optimal, but located in [5000, 6000] is a normal event. From the results, it implies that DFSPSO has a wide search domain and ability to find the optimal value is stronger than the PSO, which reduces the possibility of falling into local optimum and beneficial in finding global optimization.



For these experiments, we plot the adaption values of the final convergence into box plots, the results are as Fig. (\*\*\*), as can be seen from the figure, the results by DFSPSO are higher than PSO overall, and the box of DFSPSO is narrower, which shows that DFSPSO is more stable and better than PSO. In addition, the DFSPSO algorithm spends less than 10 seconds averagely between iteration of the feature space matrix which is far less than eigenvalue decomposition directly. After a large number of experiments, the first four steps can almost find the global CNOP in DFSPSO.

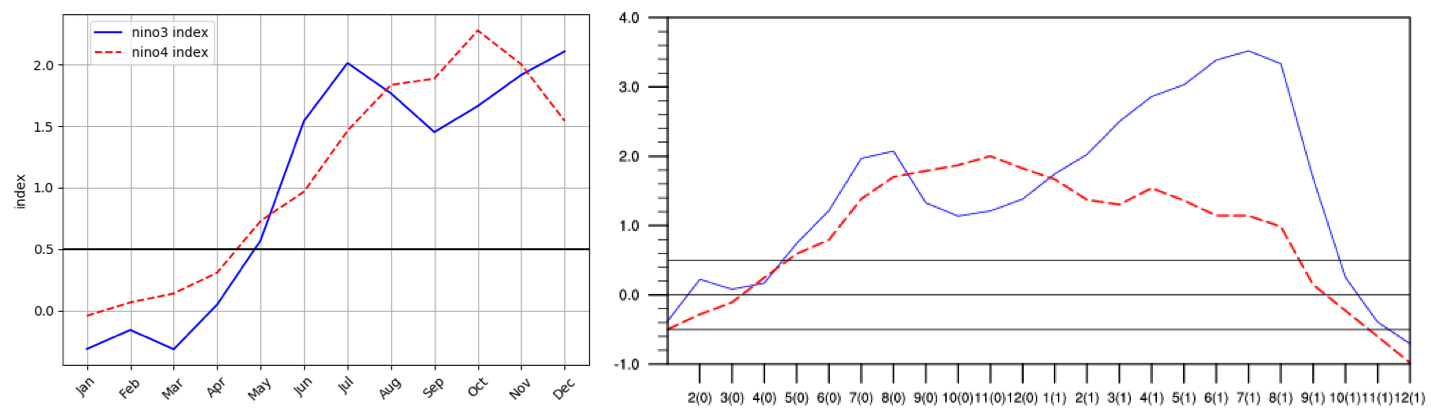
Besides, we also do some experiments on the choice of the weight of the eigenvalues. Fig. (\*\*\*) shows the relation between average convergence step and number of eigen value. It means That is to say, at the beginning of the experiment, what dimension of reduction we choose to influence the final convergence step of the algorithm. The red line is the percentage of eigen value and blue line is average convergence step of DFSPSO. From the figure we can see that the intersection is the dimension of dimension reduction chosen by our experiment. Before the intersection, the proportion of the eigenvalue is too small, and the number of convergence steps required by the algorithm is more. After the intersection, although the eigenvalue is increased, the reduction in the number of convergence steps is not obvious, the income is not high.

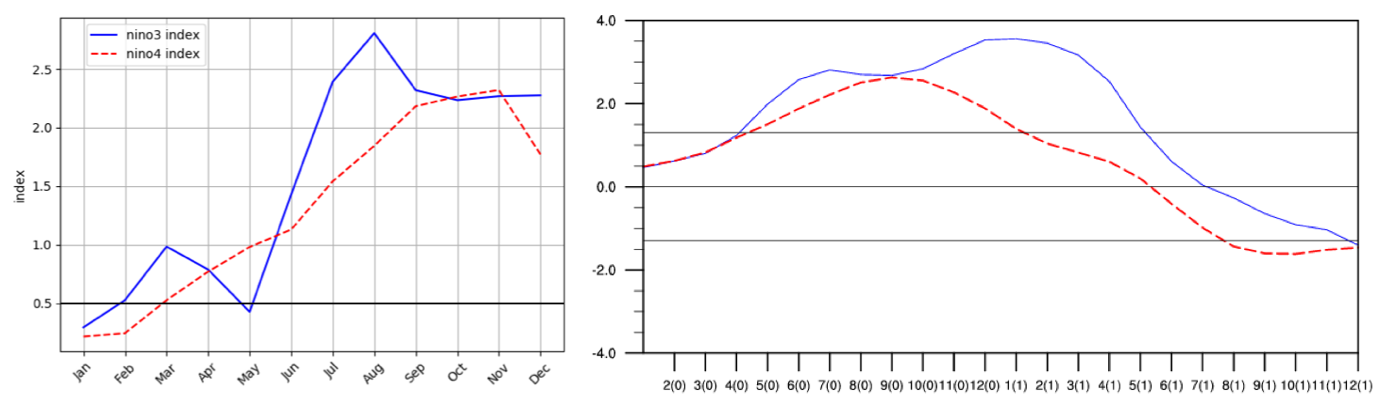


Nevertheless, the DFSPSO algorithm is a method of free of adjoint model, which makes it possible to calculate CNOP in high-resolution complex ocean-atmosphere coupling numerical models without adjoint models.

After repeating these two types of experiments respectively in some conditions: find two types of OPRs in different reference states and find them in the same reference states, two types of CNOP can be obtained successfully. The Magnitude and pattern of these two types of CNOP is almost stable. It is worth noting that, in the DFSPSO algorithm especially in CP-El Nino experiment, there exists two converge points: global CNOP and local CNOP, and the two points have different meanings in experiment. The reason for this phenomenon is that El Nino event and La Nina event always appear in pairs in nature which are both extreme climate phenomena. The adaption value of La Nina is a little bigger than CP-El Nino when setting Nino 4 region for calculating adaption, but it is much smaller than EP-El Nino when setting Nino 3 region for calculating adaption. Moreover, in the CP-El Nino experiment, the DFSPSO algorithm can always find these two CNOP successfully, but in EP-El Nino experiment, the DFSPSO algorithm usually can only find the global CNOP which represents the EP-El Nino events. This is due to in the CP-El Nino experiment, the adaption values of La Nina event and CP-El Nino event are almost the same, in the EP-El Nino experiment, the adaption value of EP-El Nino event is much larger than the adaption value of La Nina event.

In order to verify the accuracy of the algorithm, Compare the development of Nino3 and Nino4 index of the two types of El Niño events obtained by DPSPSO algorithm with the two types of El Nino events resulting from the natural running of GFDL CM, and the results are as follows





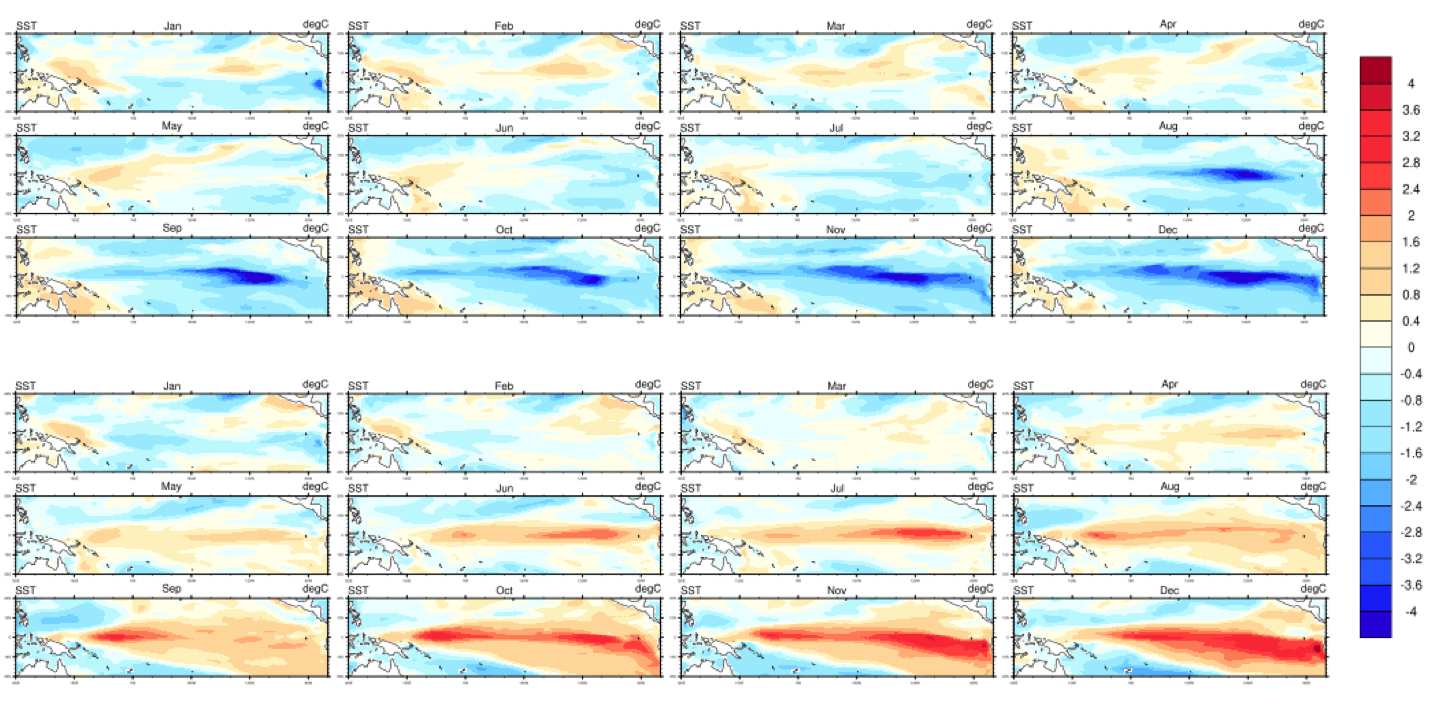
It can be seen from the Fig. (\*\*\*) that the development trend of the two types of Nino index obtained by DFSPSO algorithm are almost the same as the two types of events in GFDL CM, Moreover, it is obvious that the SSTA and SSTA evolution in these two patterns look similar with the two types of events in GFDL CM, So the results of DFSPSO algorithm are reliable.

* 1. ***Optimal precursor for EP-El Nino***

Table (\*\*\*) shows the specific parameter settings of Solving CNOP of EP-El Nino in GFDL CM

|  |  |  |  |
| --- | --- | --- | --- |
| Perturbation region | Perturbation level | Adaption function region |  |
| 30°N-30°S | 21 | Nino 3 area | 500 |

In this experiment, A normal year is chosen as the reference state, and the perturbation is added in January, Adaption function is calculated in December. Because the depth of the thermocline is the most important role in the Bjerknes positive feedback mechanism (1969) in EP-El Nino events, external tropical sea temperature signal is not very strong (参考文献), so the region of perturbation is determined between 30°N and 30°S and perturbation level is 21 including the SST and deeper under the sea surface (10m per level in GFDL CM). The CNOP is obtained as a result.



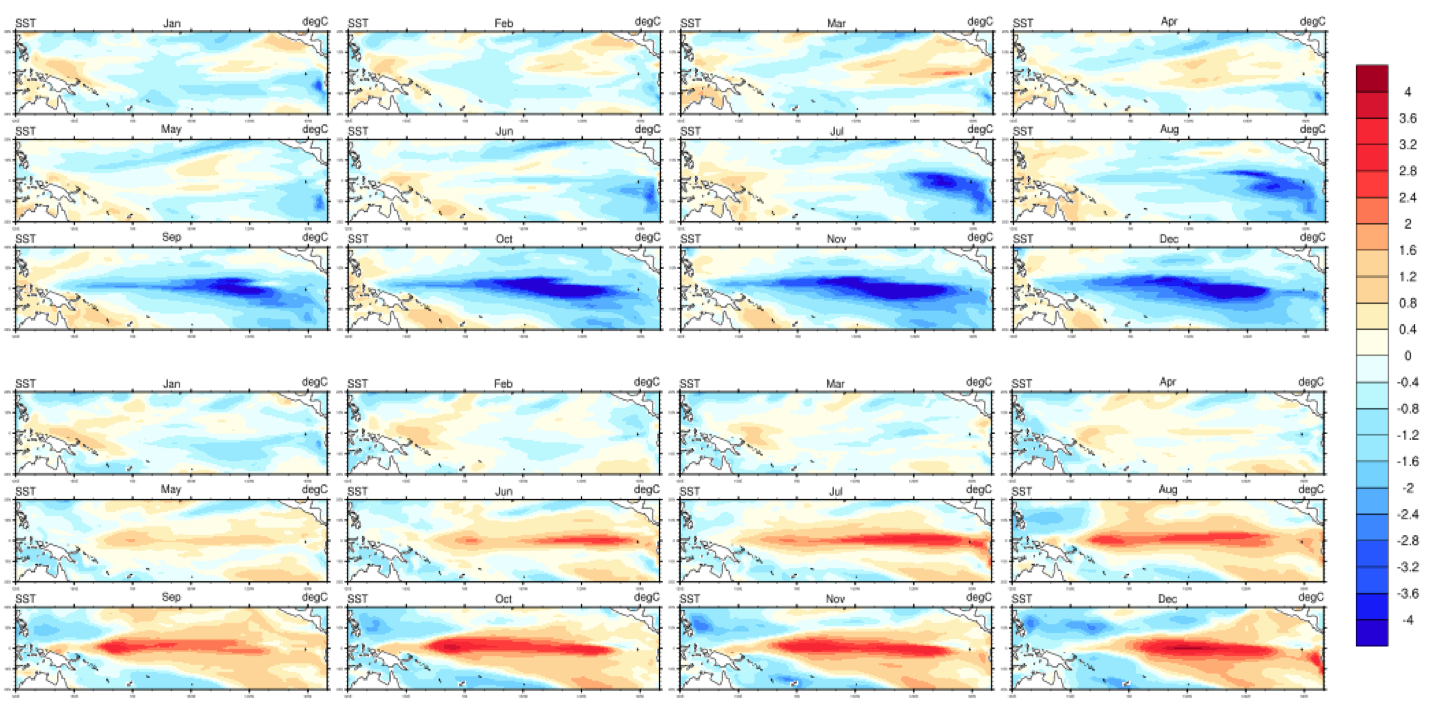
In Fig. (\*\*\*), the above is the local CNOP, which implies a La Nina Event, and the below is the global CNOP, which implies a EP-El Nino event. In the below development pattern, the positive anomalies of SST first appear in the eastern Pacific Ocean, then move toward west until touching the west coast of the Pacific and move east, finally form a large value area in the Nino3 region.

* 1. ***Optimal precursor for CP-El Nino***

Table (\*\*\*) shows the specific parameter settings of Solving CNOP of CP-El Nino in GFDL CM

|  |  |  |  |
| --- | --- | --- | --- |
| Perturbation region | Perturbation level | Adaption function region |  |
| 60°N-60°S | 11 | Nino 4 area | 480 |

In this experiment, the same normal year is chosen as the reference state, and the perturbation is added in January, Adaption function is calculated in December. Because external tropical sea temperature signal plays a more important role in CP-El Nino events than the depth of thermocline(参考文献), so the region of perturbation is determined between 60°N and 60°S and perturbation level is 11 including the SST and shallower under the sea surface than in experiment of EP-El Nino. The global CNOP and local CNOP are obtained as a result.



In Fig. (\*\*\*), the above is the global CNOP, which implies a La Nina Event, and the below is the local CNOP, which implies a CP-El Nino event. In the below development pattern, although the positive anomalies of SST are also distributed in the eastern Pacific Ocean, the positive SSTA in the Nino4 region also locally develops in the Nino 4 region, finally the SSTA large value area is in the Central Pacific Ocean.

1. **Conclusions and future works**

This propose a DFSPSO algorithm for identifying OPR for two types of El Nino events, which solve CNOP in GFDL CM. DFSPSO algorithm includes the two main ideas: dynamic feature space and Iteratively calculating for feature space. Using these two methods can speed up algorithm convergence. It is worth nothing that the DFSPSO algorithm is a method which is free of adjoint models, and at the same time, it is stable and efficient and can get better results.

In addition, the DPFPSO algorithm can be run in parallel, which can increase the search space and the possibility of obtaining the optimal solution. In the future, a more complex adaption function will be designed which may be an effective combination of more physical quantities, and in this way, it is better to find the OPRs for two types of El Nino from the perspective of physical mechanism.