

Python を用いたアンサンブル流跡線解析

田崎盛也・城間朝彰・野波秀行・前田美奈子・南雲春馬・友寄喜貴

Ensemble Analysis of Trajectories with Python

Moriya TASAKI, Tomoaki SHIROMA, Hideyuki NOHA, Minako MAEDA,
Haruma NAGUMO and Nobutaka TOMOYOSE

要旨：流跡線解析に用いられるソフトウェア HYSPLIT には流跡線の誤差評価に用いられるアンサンブル解析の機能があるが、Python 上で HYSPLIT の呼び出しと実行を行える PySPLIT ではこの機能はサポートされていない。本報告では PySPLIT のコードを修正することで Python 上でのアンサンブル解析を行うことを可能にし、Python のライブラリによりアンサンブル結果を描画し出力できるように改良した。これに加えて、アンサンブル流跡線利用による信頼楕円の作成も可能とした。これらの解析は誤差を考慮した解析結果の解釈につながるため、有用な手法であると思われる。

Key words: 流跡線解析, アンサンブル解析, 信頼楕円, Python, HYSPLIT, PySPLIT

I はじめに

流跡線解析は大気環境の分野でよく利用されており、その解析結果には誤差が含まれていることが知られている。村尾によると北半球では、移動距離のおよそ 20% の平均誤差が典型的と考えられるとされる¹⁾。これはあくまで平均であり、気象条件・解析条件によって流跡線の誤差は変動すると想定されるため、解析で得られた流跡線が信頼できるものなのか判断が難しいことが多く、何らかの方法によってこの誤差を見積もることができれば流跡線の信頼性判断の一助になると思われる。このような評価に有用な解析として NOAA (米国海洋気象局) の HYSPLIT ではアンサンブル解析の機能が実装されている。この機能はオンライン版およびデスクトップ版のどちらでも利用可能となっているが、HYSPLIT を Python 上で実行できる PySPLIT²⁻⁴⁾ではこの機能はサポートがされていない。今回、PySPLIT ライブラリのコード修正によるアンサンブル解析の実装と描画、そしてアンサンブル解析を利用した信頼楕円の作成を行ったのでこれを報告する。

II 方法

1. アンサンブル解析について

アンサンブルの手法とは、気象庁などで行われているアンサンブル予報の記述によると、初期値に小さな摂動を与えた複数の解析結果の集合 (アンサンブル) を作成し、個々の解析結果 (アンサンブルメンバー) の平均やばらつきを見ることで、解析結果の改善や不確実性を考慮することができる手法である⁵⁻⁷⁾。なお、気象分野で

は摂動を与えない解析結果をコントロールと呼ぶことがあるので、ここでも元の流跡線をコントロール流跡線とする。

HYSPLIT のアンサンブル解析はコントロールの気象データを指定した地点 (緯度, 経度, 高度) からわずかにずらした地点の値を使用する。具体的にはデフォルト設定で水平方向は 1 グリッド (GDAS では 1°), 鉛直方向は 0.01 シグマ (対流圏下層ではおよそ 100 m) を行う移動操作の組み合わせ (正方向, 移動なし, 負方向) を 3 つの方向に行うことで、コントロール流跡線を含む 27 本 ($3 \times 3 \times 3 = 27$) の流跡線が作成される。なおアンサンブル解析では気象データは異なる地点データを用いるが、開始地点は元の流跡線と同じ地点となる。

PySPLIT でアンサンブル解析を有効にするための修正箇所については次頁のコード 1 を参照いただきたい。修

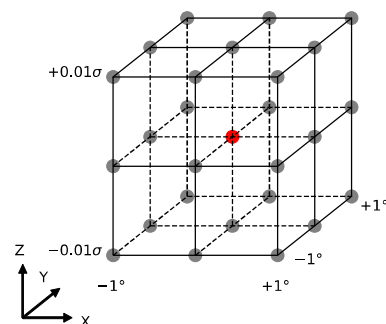


図 1. アンサンブル解析の気象グリッドイメージ。
赤い点が元の地点位置で灰色の点がアンサンブル解析のため元の地点からずらした地点位置。

正箇所 1 つ目はアンサンブル解析結果のファイル読み込みに関するものである。アンサンブル解析を行った流跡線ファイルはアンサンブルを行った 27 本の流跡線が 1 つにまとめられたファイルとなるが、PySPLIT には複数の流跡線がまとめられたファイルの個別の流跡線の読み込みにバグがあるため、その対処のための修正である。修正箇所 2 つ目は既報⁸⁾でも紹介した逆解析の機能に関するものである。逆解析はアンサンブル解析でも同様に利用可能であるが、一部に不具合がある。PySPLIT で逆解析を行う場合、流跡線ファイルの最後のエンドポイントを参照するようになっている。一方、アンサンブル解析の場合 27 番目のアンサンブルを最後のエンドポイントとして参照するため、コントロール流跡線に対する逆解析とはならない不具合を生じる。この問題に対処する

ためアンサンブル解析の場合の条件式を加えて意図した逆解析を行えるようにした。これらの修正により流跡線のアンサンブル解析が可能となる。

2. アンサンブル解析のステップ実行について

前述のアンサンブル解析は開始点の気象データに摂動を与えた場合に流跡線解析の実行結果のばらつきを示すものだが、コントロール流跡線の各エンドポイントがどの程度のばらつきをもっているかという評価は行えない。直接的にエンドポイント毎のばらつきの評価を行うにはコントロール流跡線の各エンドポイントを開始点として、繰り返しアンサンブル解析を行うステップ実行によりこれを評価することができる。

コード 1. PySPLIT ファイルの修正箇所.

```
# 修正箇所 1
# PySPLIT ライブラリファイル「hyfile_handler.py」の「_trajsplitle」関数内 226-230 行付近の以下の記述を下記のように修正する。
# 参考 URL: https://github.com/mscross/pysplit/pull/77
# この変更により複数の流跡線情報が記録されたファイルの読み込みが問題なく行えるようになる。

# 修正前
first_occurrence = [np.nonzero(sorted_indices == u)[0][0]
                    for u in unique_traj[1:]]

# 修正後
# bug reported on 2021-April-7
# first_occurrence = [np.nonzero(sorted_indices == u)[0][0]
first_occurrence = [np.nonzero(sorted_hydata[:,0] == u)[0][0]
                    for u in unique_traj[1:]]

# 修正箇所 2
# PySPLIT ライブラリファイル「trajectory_generator.py」の「_reversetraj_whileLegen」関数内 218-227 行付近の
# 以下の記述を下記のように修正する。
# この変更によりアンサンブル解析時の逆解析流跡線の開始位置がコントロール(アンサンブルではない)流跡線からアンサンブル流跡線になってしまう問題に
# 対処できる。

# 修正前
with open(trajname) as traj:
    contents = traj.readlines()

    last_timepoint = -1

    # for multiline files, critical information is in contents[-2]
    if len(contents[-1]) < len(contents[-2]):
        last_timepoint = -2

    data = contents[last_timepoint].split()

# 修正後
with open(trajname) as traj:
    contents = traj.readlines()

    last_timepoint = -1

    # for multiline files, critical information is in contents[-2]
    if len(contents[-1]) < len(contents[-2]):
        last_timepoint = -2

    if hysplit == "C:\\hysplit4\\exec\\hyts_ens":
        last_timepoint = 0
        ens_num = 0
        while ens_num != '1':
            last_timepoint = last_timepoint - 1
            ens_num = contents[last_timepoint].split()[0]

    data = contents[last_timepoint].split()
```

3. 信頼楕円の作成

前述の方法で流跡線のばらつきが評価できるようになり、アンサンブル解析の結果を利用して平均や共分散行列の算出が可能となる。これを利用し、ばらつきの目安となる信頼楕円を作成できる。信頼楕円の作成には各エンドポイントを二次元の正規分布に従うとして作成する。通常のアンサンブル解析結果で信頼楕円を作成する場合、アンサンブル流跡線の中には全く違う方向に向かうような結果がたびたび含まれることから、信頼楕円の作成に必要な正規分布のパラメータ推定値としてロバストな推定値を用い、中心位置（平均）の推定は標本平均ではなく中央値、共分散行列の推定は Minimum Covariance Determinant（最小共分散行列式、MCD）によって推定した。アンサンブル解析のステップ実行結果で信頼楕円を作成する場合は正規分布のパラメータ推定値として平均の推定は標本平均、共分散行列は標本共分散行列を用い、遡行が進むごとに誤差が増大することを考慮し、前ステップまでの共分散行列の和を加え誤差を伝播させた値を共分散行列の推定値とした。

4. 流跡線解析の設定と解析環境について

アンサンブル流跡線解析の対象となる地点と期間について、既報⁸⁾と同様に地点は沖縄県衛生環境研究所（うるま市）、期間は 2017 年度から 2020 年度の四半期ごと

に微小粒子状物質（PM_{2.5}）の成分分析対象期間である 14 日間を対象とした（表 1）。開始時間は日本標準時で 12 時、18 時、0 時、6 時の 4 つとした。流跡線遡行時間は 5 日間（120 時間）とし、気象データは GDAS を用い、出発高度は 500、1000、1500 m の 3 つとした。流跡線ファイルの作成やアンサンブル結果の流跡線解析には公式版 Python(<https://www.python.org>)を利用した解析環境を構築し、プログラミング言語 Python (3.10.9) と PySPLIT (0.3.6) を使用した。その他のライブラリについては表 2 に示す。また PySPLIT の使用にはデスクトップ版の HYSPLIT が必要となるためこれも導入しておく必要がある。アンサンブル流跡線作成のコードについては当研究所ホームページ (<https://www.pref.okinawa.jp/site/hoken/eiken/kankyo/taiki.html>) にて掲載する。前述の条件のアンサンブル流跡線作成には Intel Core i9-11900 搭載の PC で 1 時間 10 分程度の実行時間を要した。

アンサンブル解析のステップ実行では開始地点や開始時間はコントロール流跡線のエンドポイント毎の情報を参照し、流跡線遡行時間は 1 時間のみ行った。全てのコントロール流跡線に対してアンサンブル解析のステップ実行を行うと、Intel Core i9-11900 搭載の PC で 3 時間 20 分程度の実行時間を要した。

Ⅲ 結果および考察

1. アンサンブル流跡線の結果描写

アンサンブル解析によって作成した流跡線を図 2 に示す。コントロール解析の結果は既報⁸⁾と同様に色とマーカーの種類で出発高度を、線の種類で作成時間を区別できるようにした。アンサンブル解析の結果は灰色の線で表示した。図 2(a)ではアンサンブル流跡線はコントロール流跡線の周辺に多く存在している。図 2(b)ではアンサンブル流跡線はコントロール流跡線の周辺に多く存在しているが、数本のアンサンブル流跡線は全く違う方向へ向かっている。図 2(c)ではアンサンブル流跡線は様々な方向に散らばっており、コントロール流跡線の近傍にあるアンサンブル流跡線はほとんどない。図 2(a)や(b)のようにほとんどのアンサンブル流跡線がコントロール流跡線の周辺に存在していれば、結果のばらつきを考慮しても流跡線の結果は大きくは変わらないと思われるが、図 2(c)のような場合には初期値の違いが与える影響が大きいことから、流跡線の結果は信頼性が低いといったような判断ができる。

このようにアンサンブル解析の結果を参照することで初期値のわずかな違いが与える影響を考慮しながら流跡

表 1. 流跡線作成期間。

年度	春季	夏季	秋季	冬季
2017	5.10-5.24	7.24-8.7	10.23-10.27 10.30-11.9	1.18-2.1
2018	5.9-5.23	7.19-7.20 7.23-8.5	10.22-11.5	1.17-1.31
2019	5.10-5.24	7.19-7.23 7.26-8.5	10.17-10.31	1.16-1.30
2020	5.11-5.25	7.23-8.6	10.22-11.5	1.21-2.4

表 2. Python と主なライブラリのバージョン。

Name	Version
Python	3.10.9
Numpy	1.24.4
Scipy	1.14.0
scikit-learn	1.5.1
Xarray	2023.6.0
pandas	2.0.3
GeoPandas	0.13.2
Matplotlib	3.7.2
seaborn	0.13.2
japanize-matplotlib	1.1.3
Basemap	1.3.7
PySPLIT*	0.3.6*

*一部ファイルの書き換え実施（コード 1 参照）

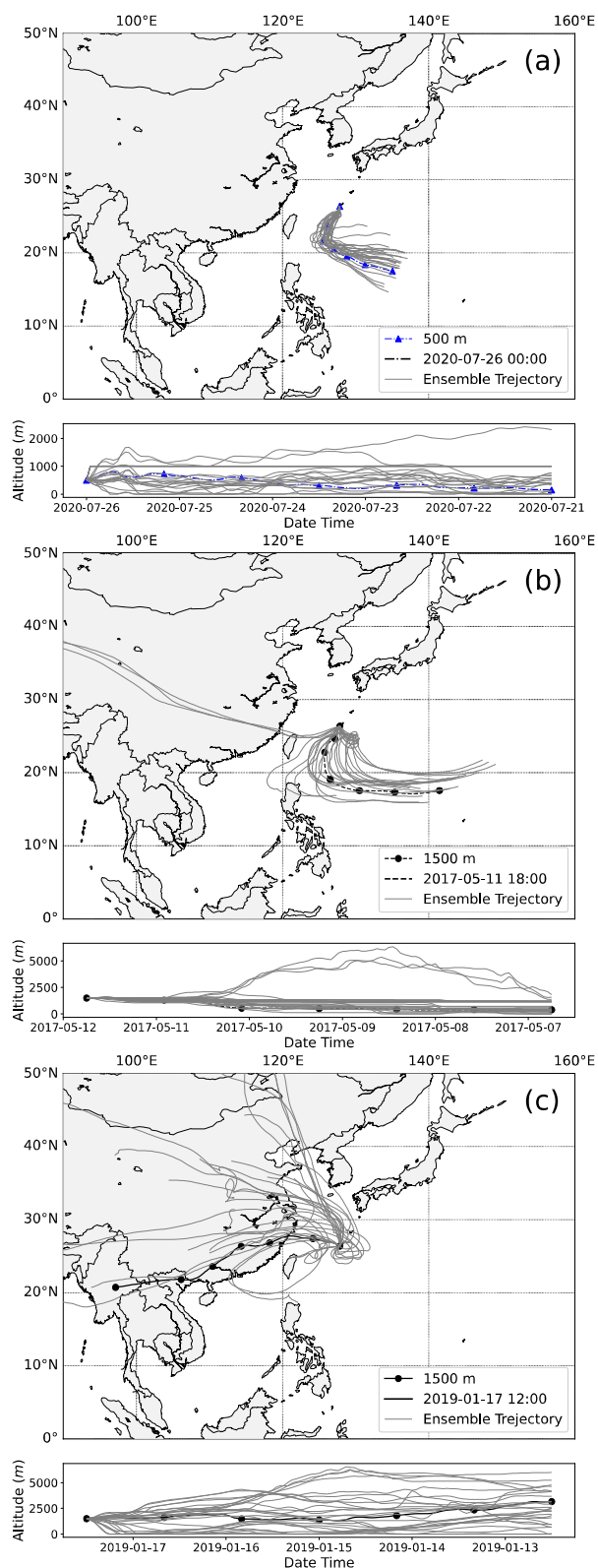


図 2. アンサンブル流跡線の一例。青色または黒色の線がコントロール流跡線，灰色の線がアンサンブル流跡線。(a)2020 年 7 月 26 日 0 時（日本標準時），出発高度 500 m。(b)2017 年 5 月 11 日 18 時（日本標準時），出発高度 1500 m。(c)2019 年 1 月 17 日 12 時（日本標準時），出発高度 1500 m。

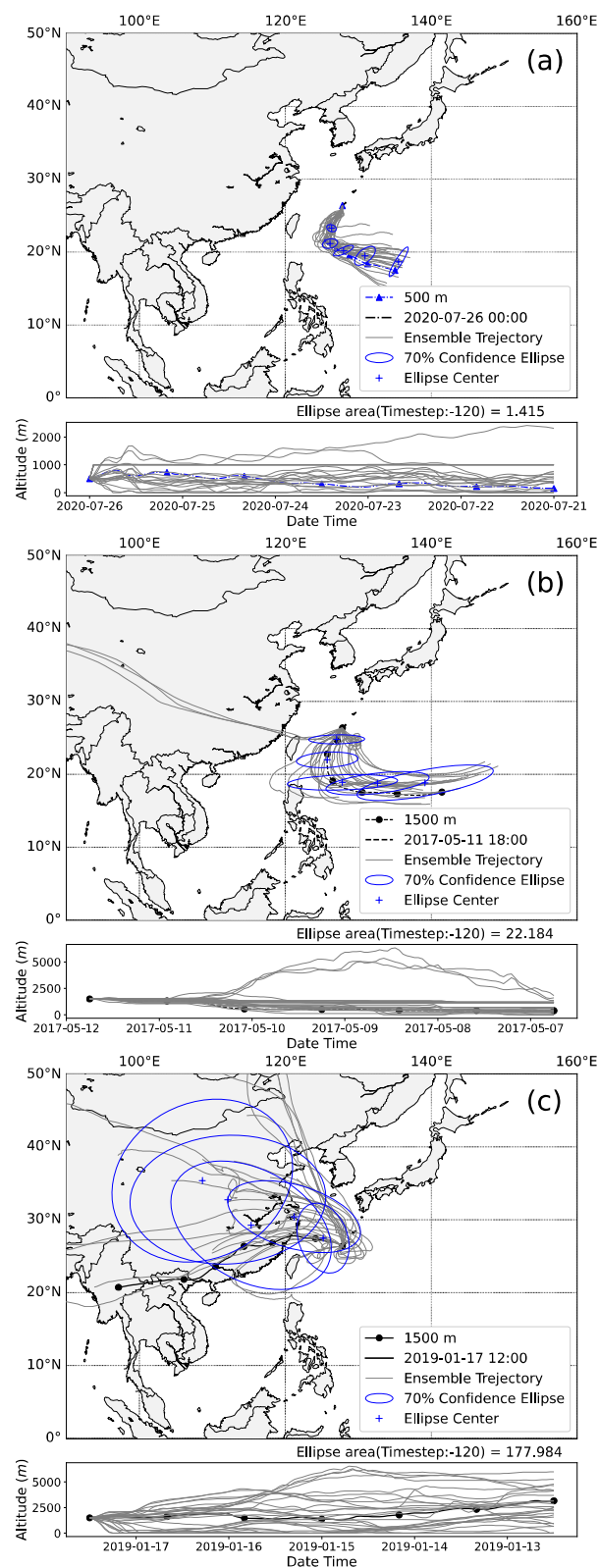


図 3. アンサンブル流跡線を利用した信頼楕円作成の一例。青色の十字が信頼楕円の中心位置，青色の楕円が 70% 信頼楕円の範囲。青色または黒色の線がコントロール流跡線，灰色の線がアンサンブル流跡線。(a), (b), (c)の開始地点，開始条件は図 2 に同じ。

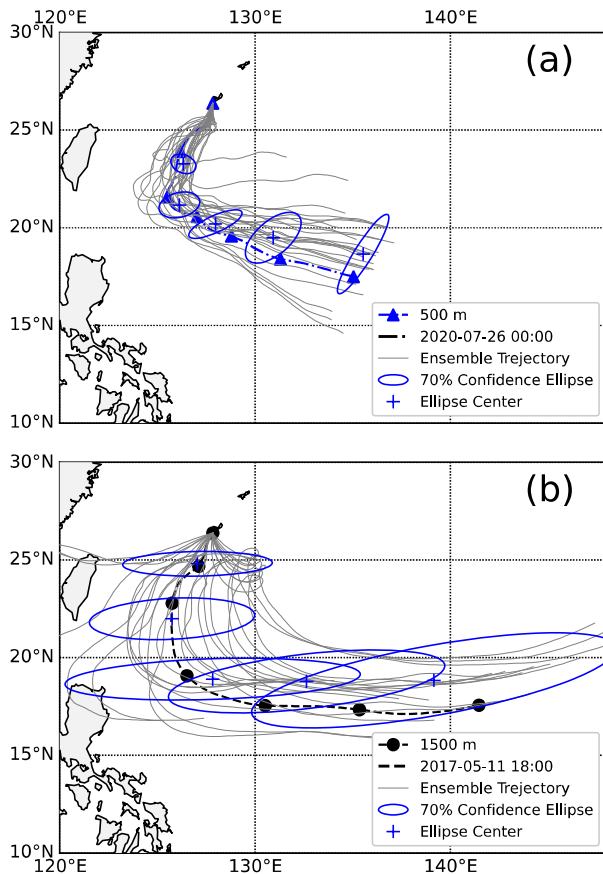


図 4. アンサンブル流跡線を利用した信頼楕円作成の一例の拡大図。青色の十字が信頼楕円の中心位置、青色の楕円が70%信頼楕円の範囲。青色または黒色の線がコントロール流跡線、灰色の線がアンサンブル流跡線。(a)、(b)の開始地点、開始条件は図3に同じ。

線の結果を解釈することが可能となる。

2. アンサンブル流跡線を利用した信頼楕円作成

前項のアンサンブル流跡線の結果に対して信頼楕円を追加したものを図3および図4に示す。図3で凡例の下に表示している「Ellipse area」の値は遡行時間最後の信頼楕円の面積（円周率×長軸×短軸）であり、長軸や短軸は共分散行列から求めている。また、信頼楕円は指定したパラメータの2次元正規分布のデータの70%が含まれる範囲の70%信頼楕円を描画しているが、この面積の算出ではデータが含まれる範囲を指定する係数は含めず計算した。図3および図4からアンサンブル流跡線のばらつきの増大や遡行時間が進むにつれて信頼楕円の範囲や面積が大きくなるのわかる。また、信頼楕円の中心

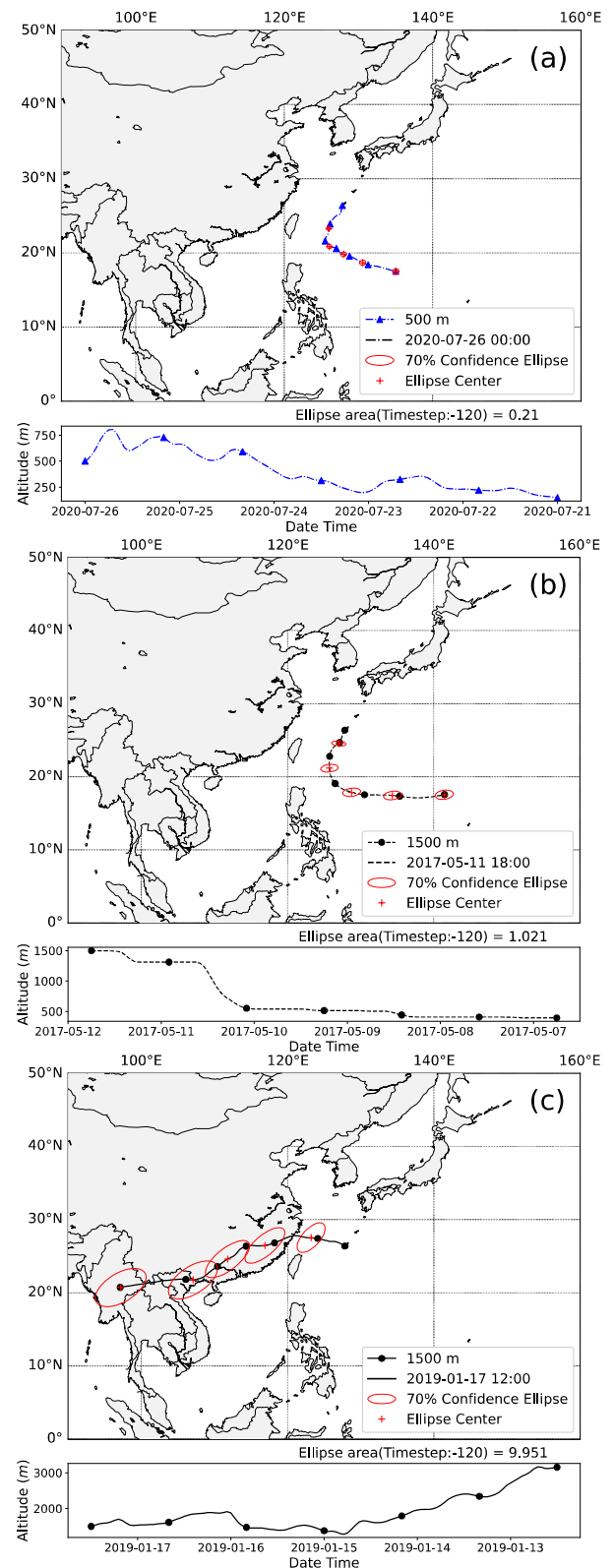


図 5. アンサンプルのステップ実行による信頼楕円作成の一例。赤色の十字が信頼楕円の中心位置、赤色の楕円が70%信頼楕円の範囲。青色または黒色の線がコントロール流跡線。(a)、(b)、(c)の開始地点、開始条件は図2および図3に同じ。

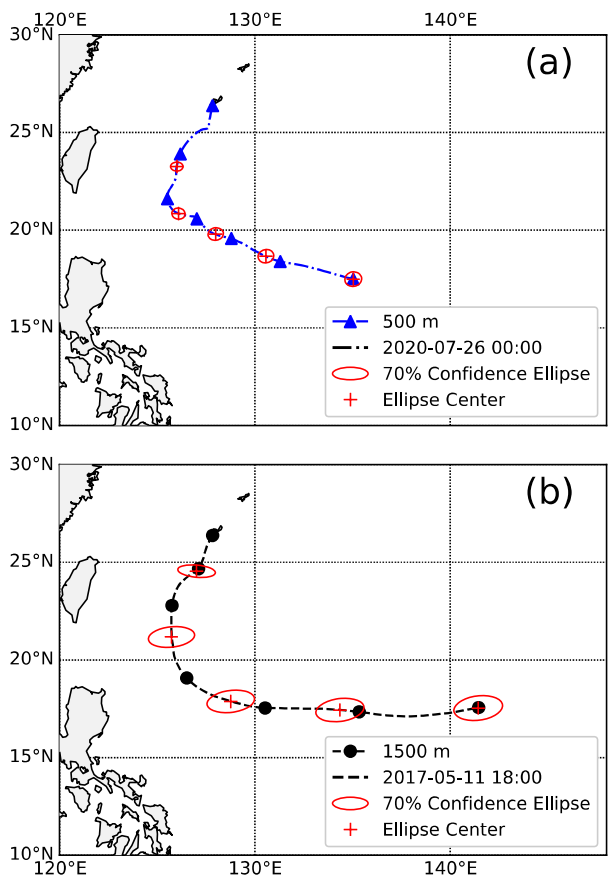


図 6. アンサンブルのステップ実行による信頼楕円作成の一例の拡大図. 赤色の十字が信頼楕円の中心位置, 赤色の楕円が70%信頼楕円の範囲. 青色または黒色の線がコントロール流跡線. (a), (b)の開始地点, 開始条件は図 5 に同じ.

位置はコントロール流跡線とは一致しておらず, この方法で作成した信頼楕円はアンサンブル流跡線のばらつきは示せてもコントロール流跡線のエンドポイントの誤差を示すような評価には適当でないこともわかる.

3. アンサンブルのステップ実行による信頼楕円作成

アンサンブルのステップ実行による結果に対して信頼楕円を作成したものを図 5 および図 6 に示す. 図 5 で凡例の下に表示している「Ellipse area」の値は図 3 と同様に遡行時間最後の信頼楕円の面積であり, 信頼楕円の範囲も 70%信頼楕円を描画している. 図 5 と図 6 では計算方法の違いから信頼楕円の大きさは前項の手法よりも小さくなっている. また遡行時間が進むことにより, 誤差の伝播を考慮していることから信頼楕円のサイズが大きくなっていることが読み取れる. また, 前項の手法とは違い信頼楕円の中心位置はコントロール流跡線とほぼ一致しており, エンドポイントの誤差を示す評価法として

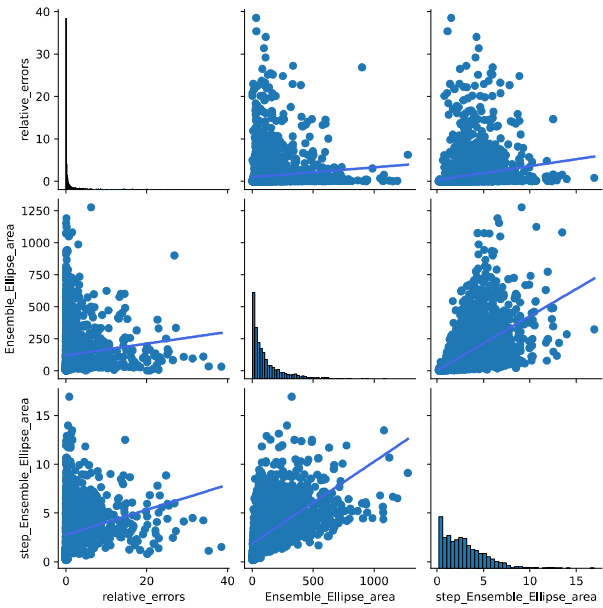


図 7. 各指標の関係を表す散布図. 対角線上のグラフは各指標のヒストグラムを表す. 軸タイトルの「relative_errors」は相対誤差, 「Ensemble_Ellipse_area」はアンサンブル流跡線を利用した信頼楕円の面積, 「step_Ensemble_Ellipse_area」はアンサンブルのステップ実行による信頼楕円の面積を表す.

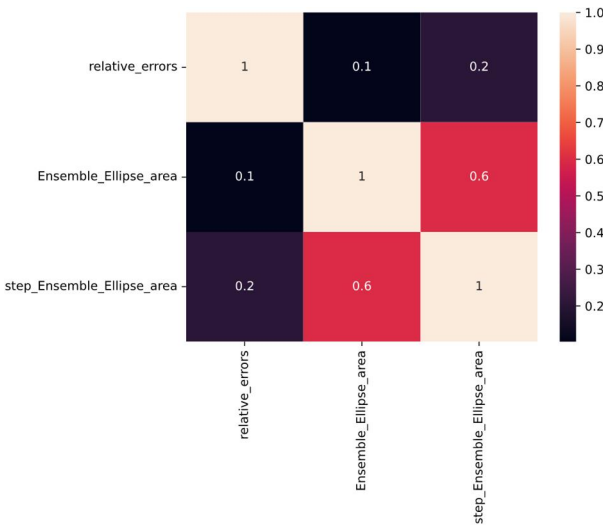


図 8. 各指標の相関関係を表すヒートマップ. 軸タイトルは図 7 に同じ.

妥当であると思われる.

4. 各指標の相関関係

既報で取り上げた相対誤差⁸⁾, アンサンブル流跡線による信頼楕円の面積, アンサンブルのステップ実行による信頼楕円について, 散布図とヒートマップにより相関

を確認した (図7および図8). 相対誤差は信頼楕円の面積との相関は 0.1 または 0.2 と低い, 信頼楕円の面積間の相関は 0.6 とやや高い結果となった. 信頼楕円の面積はどちらの手法でも細かい計算法は違うもののアンサンブル解析の結果を利用して, 解析開始点気象データをその近傍のデータに変更し使用することで結果に攪乱を与えるため, 気象データに起因する誤差を評価しているのに対し, 相対誤差は逆解析流跡線の利用で積分誤差を評価していることから, 評価対象の違いのため相関が低いと考えられる. 流跡線の誤差評価を行う上では積分誤差の評価を相対誤差で行い, 気象データに由来する誤差を評価するには目的に応じてアンサンブル流跡線による信頼楕円の面積またはアンサンブル解析のステップ実行による信頼楕円の面積により誤差評価を行うことが妥当であると思われる.

IV まとめ

PythonのライブラリであるPySPLITのコードの修正を行うことにより, HYSPLITの機能にある流跡線のアンサンブル解析をPythonで行うことを可能とした. これにより, 流跡線解析の誤差をアンサンブル流跡線の描画やそれを利用した信頼楕円の作成により見積もることができた. これらの解析は誤差を考慮した解析結果の解釈につながるため, 有用であると思われる.

V 参考文献

- 1) 村尾直人 (2011) 大気モデル—第6講 流跡線解析—, 大気環境学会誌, 46, A61—A67.
- 2) Cross, M., PySPLIT: A Package for the Generation

Analysis and Visualizations of HYSPLIT Air Parcel Trajectories, Proc. 14th Ann. Scientific Computing with Python Conf. (SciPy 15), 2015.

- 3) Warner, M. S. C., Warner. Introduction to PySPLIT: A Python toolkit for NOAA ARL's HYSPLIT model. Computing in Science and Engineering, 20(5), 47–62.
- 4) Warner, M. S. C., PySPLIT. < <https://github.com/mscross/pysplit>>. 2024年7月アクセス.
- 5) 気象庁. アンサンブル予報. <<https://www.jma.go.jp/jma/kishou/known/whitep/1-3-8.html>>. 2024年7月アクセス.
- 6) 気象庁. 予測に伴う誤差とアンサンブル予報. <https://www.jma.go.jp/jma/kishou/known/kisetsu_riyou/method/ensemble.html>. 2024年7月アクセス.
- 7) 気象庁. 数値予報解説資料集(令和5年度) 1.5 アンサンブル予報. <https://www.jma.go.jp/jma/kishou/books/nwpkaisetu/latest/1_5.pdf>. 2024年7月アクセス.
- 8) 田崎盛也 (2022) Pythonによる流跡線解析. 沖縄県衛生環境研究所報, 56: 78–101.

VI 付録

本報告に関連するコードについて, 本文中でも紹介したPySPLITファイルの修正箇所コードのほか, アンサンブル流跡線作成や描画, 信頼楕円の作成・描画のコードを当研究所ホームページ (<https://www.pref.okinawa.jp/site/hoken/eiken/kankyo/taiki.html>) にて掲載する.