Clustering Stochastic Weather Scenarios using Influence Model-based Distance Measures

Chenyuan He* and Yan Wan†
University of Texas at Arlington, Arlington, TX, 76019

Off-nominal weather conditions are the leading causes of air traffic delays in the National Airspace System (NAS). To minimize traffic delays and meanwhile guarantee aviation safety, a weather data-driven air traffic management decision solution is important. This decision solution requires a fast and accurate approach to cluster weather scenarios. Based on retrieved offline management solutions corresponding to similar weather scenarios, management solutions can be quickly designed. In this paper, we use the influence model to capture stochastic spaiotemporal weather spread, and develop model-based distance measures for stochastic weather spread. We compare these model-based distance measures with a data-driven distance measure for spatiotemporal scenario clustering using simulation studies. Since the performance of a data-driven distance measure can be easily affected by the selection of multiple parameters, the model-based distance measures have advantages such as improved robustness and less user interference.

I. Nomenclature

Y_i	=	a spatiotemporal weather scenario i
B	=	the set of spatial cells
b_n	=	a specific spatial cell
T	=	the set of time points
t_k	=	a specific time point
$I_{i,n,k}$	=	intensity of scenario Y_i at spatial cell b_n and time point t_k
$\phi_{n,w}$	=	a size-w spatial window centered at b_n and contains all the spatial cells within $(w-1)$ hops to b_n
$\phi_{k,h}$	=	a size-h temporal window starting from t_k and contains itself and the subsequent $(h-1)$ time points
$\mathcal{D}_{i,j,w,h}$	=	the distance of spatial resolution w and temporal resolution h between spatiotemporal scenarios Y_i and Y_j
$\beta_{r,l}$	=	an importance weighing factor of the spatial cell b_r at time point t_l
Φ_w	=	the full set of spatial windows of size w
Φ_h	=	the full set of temporal windows of size h
$\hat{I}_{i,n,k}$	=	scaled intensity of $I_{i,n,k}$
$\lambda_{n,w}$	=	spatial contribution factor
$\tau_{k,h}$	=	temporal contribution factor
$\mathcal{D}_{i,j}$	=	the total distance between spatiotemporal scenarios Y_i and Y_j
w_{max}	=	maximum spatial window
h_{max}	=	maximum temporal window
σ_w	=	spatial weighting factor
α_h	=	temporal weighting factor
$\mathcal{D}$	=	the overall distance matrix of multiple scenarios
N	=	the number of regions of an area in an influence model
D	=	network influence matrix
d_{ij}	=	probability that region i is influenced by region j
A_{ij}	=	transition matrix of the local Markov chain between region i and region j
M_i	=	the number of weather statuses for region i
a_{mn}	=	the mth row and nth column entry of A_{ij}
M	=	the number of weather statuses for each region in a homogeneous influence model

*Ph.D. Student, Department of Electrical Engineering.
†Associate professor, Department of Electrical Engineering, AIAA Senior Member.

A	=	transition matrix of the local Markov chain in a homogeneous influence model
$S_i[k]$	=	status of region i at time k in vector-form
$s_i[k]$	=	status of region i at time k in scalar-form
$S[k]$	=	the whole area's weather state matrix at time k
$s[k]$	=	state of the master Markov chain G at time k
$p_i[k]$	=	probability mass function (PMF) for the weather status of region i at time step k
$p[k]$	=	PMF of the whole area's weather state at time k
G	=	master Markov chain mapped from an influence model
g_{mn}	=	transition probability from master Markov state m to n
O	=	an intermediate matrix for the calculation of D from G
V	=	an intermediate matrix for the calculation of D from G
o_{ij}	=	the ith row and jth column entry of O
v_{ij}	=	the ith row and jth column entry of V
$L(\theta_i; Y_i)$	=	the likelihood of scenario Y_i given θ_i
$\hat{\theta}_i$	=	the estimation of θ_i from Y_i
w_{ij}	=	the likelihood of scenario Y_j given $\hat{\theta}_i$
$\mathcal{D}^{SS}$	=	the distance matrix for the simple symmetric distance measure
$\mathcal{D}^{BP}$	=	the distance matrix for the BP metric
$\mathcal{D}^{YY}$	=	the distance matrix for the Yin-Yang distance measure

II. Introduction

Off-nominal weather conditions, such as low ceiling, precipitation, wind, fog, and icing conditions are the leading causes of air traffic delays in the National Airspace System (NAS), and hence play a significant role in air traffic management [1, 2]. Strategic air traffic management is concerned with allocating limited resources in the airspace 2-8 hours ahead of time, in preparation for weather and other off-nominal conditions [3, 4]. To minimize traffic delays and meanwhile guarantee aviation safety, a weather-driven air traffic management decision solution is important. An advisable approach is to construct a database of weather scenarios and their corresponding management solutions. Based on the retrieved offline management solutions corresponding to similar weather scenarios from the database, management solution for the current weather scenario can be quickly designed [5]. For example, a spatiotemporal scenario data-driven decision framework was developed and its efficiency was verified through an application study on multi-UAV path planning [6].

Weather scenarios have stochastic spatiotemporal spread properties in the strategic time frame [7].The aforementioned traffic management decision framework requires clustering algorithms to classify weather scenarios. In the literature, clustering algorithms can be broadly classified into two categories, model-based and data-based [8]. Both of them require distance measures to quantify the similarity between pairwise scenarios. For model-based clustering algorithms, distance measures are applied to the models underlying data, while for data-based clustering algorithms, distance measures are applied directly to the scenario data. In our previous study, we analyzed the spatiotemporal correlations of weather scenarios, and developed a data-driven multi-resolution spatiotemporal distance measure to cluster spatiotemporal spread patterns [7]. The performance of this method was verified using plenty of examples and real NAS weather-impact datasets. The method has a number of parameters to configure and the performance of the clustering accuracy relies on proper selections of these parameters. Parameter selection guidelines were provided in [7] based on knowledge of the spread and sensitivity studies of these parameters.

In this paper, we revisit the data-driven multi-resolution distance measure approach, use it as a benchmark, and design model-based distance measures that are more robust and require less user interference. Our development for the model-based distance measures is composed of two steps. The first step is to extract the underlying model dynamics from spatiotemporal scenario data, and the second step is to calculate the distance between pairwise scenarios based on the estimated model parameters. Examples of models used in the literature include regression models, auto regressive moving average (ARMA) models and auto regressive integrated moving average (ARIMA) models. We here use the influence model, a discrete-time stochastic model to capture spatiotemporal weather spread processes [9]. The influence model has nice characteristics, such as reduced-order representation and computational efficiency.

Various model-based distance measures have been studied in the literature. Papers [10–13] developed model-based distance measures to cluster hidden Markov model (HMM) driven scenarios. Paper [11] was one of the earliest work that developed a probabilistic distance measure based on the Kullback-Leibler (KL) number. In [12], a probabilistic

model-based distance measure was used for the unsupervised classification of EEG signals. In [13], a distance measure that considers the cross-fitness of two scenarios was developed to cluster time series data of variable length, noisy and multiple dimensions. In [10], a model-based distance measure embeds the information of the whole dataset into each pairwise distance for sequential data clustering. Per knowledge of the authors, this paper is a first attempt to develop influence model-based distance measures for the clustering of stochastic spatiotemporal senario data.

The rest of this paper is organized as follows. Section III reviews the data-driven multi-resolution spatiotemporal distance measure, which serves as the benchmark for this paper, and formulates the clustering problem. In Section IV, we introduce the influence model, its estimation related properties and the estimate algorithms. We then introduce three model-based distance measures for the influence model. Integrating the distance measures and the hierarchical clustering algorithm, we construct the influence model-based clustering algorithm. Section V uses simulation studies to compare and analyze the data-driven distance measure and three model-based distance measures.

III. Literature Review and Problem Formulation

A. Data-Driven Multi-Resolution Spatiotemporal Distance Measure

In our previous work, a data-driven multi-resolution spatiotemporal distance measure approach was developed to group spatiotemporal scenarios [7]. The measure adopts the concept of moving windows to scan scenarios with increasingly coarser resolutions along both temporal and spatial dimensions, and calculates the similarity between pairwise scenarios by summing the difference for all resolutions. A distance matrix can be obtained by iterating the algorithm for all the pairs of scenarios. Combining the distance matrix with standard distance-based clustering algorithms, such as hierarchical clustering in our case, the spatiotemporal scenarios can be classified into groups. The data-driven multi-resolution distance measure approach is efficient in clustering spatiotemporal scenarios with specific spread patterns. In addition, the algorithm is applicable not only to regular-shaped spatial cells, but also to randomly-shaped spatial cells. Other features include the correction of boundary effects and the permission of heterogeneous contributions of spatial cells and time points. The performance of the data-driven multi-resolution distance measure approach has been verified with real NAS weather-impact dataset.

The algorithm of the data-driven multi-resolution spatiotemporal distance measure is summarized as follows. Let Y_i and Y_j denote two spatiotemporal scenarios, each of which is composed of the same number of spatial cells and temporal length. Each scenario is captured by collecting snapshots of all spatial cells at all time points. Let B denote the set of spatial cells and $b_n \in B$ refer to a specific spatial cell. Similarly, let T denote the set of time points and $t_k \in T$ refer to a specific time point. The intensity of Y_i at a spatial cell b_n and time point t_k is denoted by $I_{i,n,k} \geq 0$.

Two moving-windows, one for scanning spatial cells and the other one for scanning time points, are denoted as $\phi_{n,w}$ and $\phi_{k,h}$ respectively. $\phi_{n,w}$ is a size-w window centered at the spatial cell b_n and contains all the spatial cells within $w-1$ hops to b_n. $\phi_{k,h}$ is a size-h window starting from the time point t_k and contains itself and the subsequent $h-1$ time points. The distance of spatial resolution w and temporal resolution h between the two scenarios $\mathcal{D}_{i,j,w,h}$ is calculated by comparing the aggregated intensities with fixed spatial window size w and temporal window size h.

$$\mathcal{D}_{i,j,w,h} = \sum_{\phi_{n,w} \in \Phi_w} \sum_{\phi_{k,h} \in \Phi_h} \frac{1}{|\phi_{n,w}||\phi_{k,h}||\Phi_h|} \left| \sum_{b_r \in \phi_{n,w}} \sum_{t_l \in \phi_{k,h}} \frac{\hat{I}_{i,r,l}}{\lambda_{r,w}\tau_{l,h}} - \sum_{b_r \in \phi_{n,w}} \sum_{t_l \in \phi_{k,h}} \frac{\hat{I}_{j,r,l}}{\lambda_{r,w}\tau_{l,h}} \right|, \tag{1}$$

where

$$\begin{aligned} \lambda_{n,w} &= \sum_{\phi_{r,w} \in \{\phi_{r,w} | b_n \in \phi_{r,w}\}} \frac{1}{|\phi_{r,w}|} \\ \tau_{k,h} &= \sum_{\phi_{l,h} \in \{\phi_{l,h} | t_k \in \phi_{l,h}\}} \frac{|T|}{|\phi_{l,h}||\Phi_h|} \end{aligned}$$

where $|\cdot|$ denotes the cardinality, Φ_w denotes the full set of spatial windows of size w, and Φ_h denotes the full set of temporal windows of size h. $\hat{I}_{i,r,l} = \beta_{r,l} I_{i,r,l}$ is a scaled intensity with $\beta_{r,l} > 0$ weighing the importance of the spatial cell b_r at time point t_l. In our study, we set $\beta_{r,l} = 1$ for all spatial cells and time points. $\lambda_{r,w}$, a spatial contribution factor, is used to correct the boundary effect of the spatial cells, so that each spatial cell contributes equally to the distance calculation. The temporal contribution factor $\tau_{l,h}$ functions in a similar way.

After iterating all the spatial windows and temporal windows, the total distance between scenarios Y_i and Y_j can be computed as

$$\mathcal{D}_{i,j} = \sum_{h=1}^{h_{max}} \sum_{w=1}^{w_{max}} \mathcal{D}_{i,j,w,h} \frac{\delta_w \alpha_h}{\sum_{h=1}^{h_{max}} \sum_{w=1}^{w_{max}} \delta_w \alpha_h}, \tag{2}$$

where w_{max} and h_{max} denote the sizes of the maximum spatial window and temporal window respectively. With the increasing of window size, the resolution decreases. $\delta_w > 0$ and $\alpha_h > 0$ are weighting factors for spatial window and temporal window respectively. In general, larger window size contributes less to the calculation of distance due to its coarse resolution. Here we select δ_w and α_h to be negative exponential functions, i.e., $\delta_w = e^{-\sigma(w-1)}$ and $\alpha_h = e^{-\rho(h-1)}$, where σ, $\rho \geq 0$.

Repeating the procedures for each pair of scenario, a distance matrix can be obtained. The overall procedure to calculate the distance matrix for multiple scenarios is summarized as follows.

Algorithm 1 Data-Driven Multi-Resolution Distance Measure Algorithm

Input:

Multiple scenarios $Y = \left[Y_1, Y_2, \cdots, Y_L\right]$.

Output:

Distance matrix $\mathcal{D}$.

1: **for** pair of scenarios Y_i and Y_j **do**
2: **for** pair of spatial resolution $w = 1 : w_{max}$ and temporal resolution $h = 1 : h_{max}$ **do**
3: Calculate the distance $\mathcal{D}_{i,j,w,h}$ for fixed spatial window size w and temporal window size h according to (1).
4: **end for**
Calculate the total distance $\mathcal{D}_{i,j}$ between scenarios Y_i and Y_j using Equation (2).
5: **end for**

B. Problem Formulation

The performance of the data-driven multi-resolution spatiotemporal distance measure depends highly on the selection of parameters. To achieve accurate clustering results, one may need to understand the spatial graph structure and spatiotemporal spread dynamics to select appropriate parameters, which can be challenging for users. We study in this paper the model-based distance measures which have improved robustness and require less user interference. The procedure of the model-based distance measure clustering approach islisted as follows.

Step 1: Estimate the model parameters underlying each spatiotemporal scenario.

Step 2: Apply model-based distance measures directly to the model parameters between pairwise scenarios and obtain a distance matrix for all the spatiotemporal scenarios.

Step 3: Apply standard distance-based clustering algorithms to the distance matrix to group the spatiotemporal scenarios.

To facilitate these three steps, our problems can be formulated as follows.

Problem 1: Find a model-based distance measure to facilitate clustering.

Problem 2: Find a model which is capable to capture spatiotemporal weather dynamics so that we can apply the model-based distance measure found in *Problem 1* to the model-driven scenarios and analyze the performance.

IV. Influence Model-Based Distance Measures

We use the influence model, a discrete-time stochastic model to capture the spatiotemporal spread of weather scenarios. We first introduce the influence model. Then we study its estimation related properties, and propose two estimation algorithms. One estimator is the widely used maximum likelihood estimator (MLE), and the other one called linear-algebra-based influence model estimator is derived based on the unique properties of the influence model [14]. Three KL divergence based distance measures are introduced and applied to the influence model. Combining them with hierarchical clustering algorithm, we construct the framework for influence-model based clustering algorithm.

A. Overview of Influence Model

The influence model is a discrete-time stochastic model that succinctly captures uncertain spatiotemporal spread dynamics. It describes the evolution of weather statuses of an area of multiple regions according to their interactions. At any time step k, a region randomly picks one of its neighbors (including itself) as its determining region, and refreshes its status at time $k+1$ based on the status of the determining region.

For an area composed of N regions, a network influence matrix $D \in R^{N\times N}$ is used to capture the network interaction among regions. A local Markov chain $A_{ij} \in R^{M_i\times M_j}$ is used to capture the local Markov process between a pair of regions i and j, where $i, j \in \{1, 2, \cdots, N\}$, and M_i and M_j denote the number of weather statuses of region i and region j respectively. Both D and A_{ij} are right stochastic matrices. The entry d_{ij} denotes the probability that region i is influenced by region j. It is natural to assume that the probability becomes smaller with the increase of distance between region i and j. Each entry a_{mn} of A_{ij} denotes the probability that region i will be in weather status m the next time step when region j is in weather status n at the current time step. In this paper, we focus on a special class of influence model, whose name is homogeneous influence model, where the numbers of weather statuses are the same for any region and the local Markov chains for any pairwise regions are the same as well. Hence, a homogeneous local Markov chain $A \in R^{M\times M}$ can be used to capture the local influences, where M denotes the number of weather statuses of each region. In the rest of this paper, the homogeneous influence model is referred to as influence model when it does not cause confusion.

A simple example is shown in Figure 1, the area is composed of 3 regions, and each region has 2 weather statuses, normal and cold. For example, the first row of D denotes how region 1 is influenced by other regions (including itself), and the other rows function in a similar way. The local Markov chain A is able to capture both positive and negative influence between a pair of regions. Here positive influence means region i has a tendency to follow its neighbor's weather status, and negative influence means region i has a tendency to reject its neighbor's weather status. The first row of A shows a positive influence and the second row shows a negative influence.

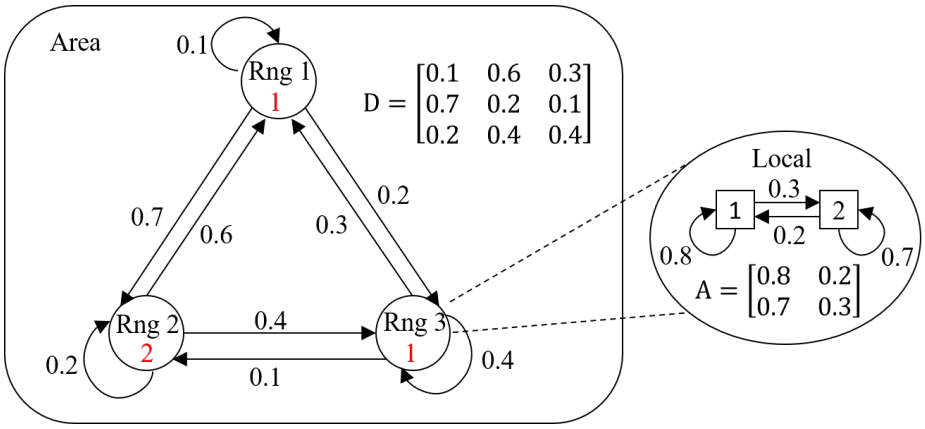

Fig. 1 An influence model example of 3 regions and each region has two weather statuses, with '1' denoting normal and '2' denoting cold.

We use a scalar $s_i[k] \in \{1, 2, \cdots, M\}$ to denote the weather status of region i at time k. Let also a length-M row vector $S_i[k]$ to represent the vector form of region i's weather status at time k, where $S_i[k]$ is filled with '0's except a value '1' at the position $s_i[k]$. $s_i[k]$ and $S_i[k]$ has a one-to-one mapping relationship. The weather state $S[k]$ of the whole area can be described by cascading $S_i[k]$ for all region i,

$$S[k] = \left[S_1^T[k], S_2^T[k], \cdots, S_N^T[k]\right]^T. \tag{3}$$

where the superscript T denotes the transpose operation.

Similarly, a length-M row vector $p_i[k]$ is used to represent the probability mass function (PMF) for the weather status of region i at time step k. The PMF of the whole area's weather state can be represented by a state probability matrix $p[k]$ through cascading $p_i[k]$ for all i,

$$p[k] = \left[p_1^T[k], p_2^T[k], \cdots, p_N^T[k]\right]^T. \tag{4}$$

Based on the evolution rules of the influence model, we have

$$p[k+1] = DS[k]A. \tag{5}$$

The weather state of the area at the next time step $k+1$ is realized according to $p[k+1]$ as

$$S[k+1] = Realize(p[k+1]). \tag{6}$$

The influence model preserves Markov property since the next weather state of the area only depends on the current state but nothing else from the past according to (5) and (6). It is a reduced-order representation of a master Markov chain G with M^N states. We use a scalar $s[k]$ ranging from 1 to M^N to represent the state of G at time k, and the mapping from $s_i[k]$ to $s[k]$ is $s[k] = \sum_{i=1}^{N}(s_i[k]-1)M^{N-i}+1$. Conversely, for $s[k] = n$ we have $s_i[k] = n_i$, where n_i denotes the weather status of region i which is consistent with the overall area state n. The corresponding $S_i[k] = \boldsymbol{e_{n_i}}$, where $\boldsymbol{e_{n_i}}$ is a row vector with '1' at n_ith position and otherwise filled with '0's.

For each entry of the master Markov chain G, we use g_{mn} to denote the transition probability from $s[k] = m$ to $s[k+1] = n$. g_{mn} can be computed as the product of all sites' conditional probabilities.

$$\begin{aligned} g_{mn} &= P(s[k+1] = n | s[k] = m) \\ &= P(s_1[k+1] = n_1, \cdots, s_N[k+1] = n_N | s_1[k] = m_1, \cdots, s_N[k] = m_N) \\ &= P(S_1[k+1] = \boldsymbol{e_{n_1}}, \cdots, S_N[k+1] = \boldsymbol{e_{n_N}} | S_1[k] = \boldsymbol{e_{m_1}}, \cdots, S_N[k] = \boldsymbol{e_{m_N}}) \\ &= \prod_{l=1}^{N} P(S_l[k+1] = \boldsymbol{e_{n_l}} | S_1[k] = \boldsymbol{e_{m_1}}, \cdots, S_N[k] = \boldsymbol{e_{m_N}}) \\ &= \prod_{l=1}^{N} (\sum_{r=1}^{N} d_{lr} \boldsymbol{e_{m_r}} A) \boldsymbol{e_{n_l}^T}. \end{aligned} \tag{7}$$

Compared with the master Markov chain G which has M^N states, the influence model has only $M \times N$ states at each time step, which facilitates a tractable analysis of network dynamics with computational efficiency, and having broad usage in designing, modeling, and analyzing complex spatiotemporal scenarios.

B. Estimation Related Properties of the Influence Model

We study the properties of the influence model related to the estimation of model parameters from data. Specifically, the master Markov chain G can be uniquely determined from a large volume of scenario data based on the law of large numbers, and the relationship of the reverse mapping from the master Markov chain G to the local Markov chain A and the network influence matrix D is summarized as follows[14].

a). Given a master Markov chain G constructed from a network influence matrix D and a local Markov chain A, A can be recovered from G as

$$a_{mn} = \sqrt[N]{g_{l_m, l_n}}, \tag{8}$$

where l_m represents the state of G where all the regions are in the influence model weather status m. (8) can be obtained directly from (7).

b). Given a master Markov chain G constructed from a network influence matrix D and a local Markov chain A, D can be recovered from G as $vec(D^T) = V(O^T O)^{-1} O^T vec(G^T)$ if matrix O has full column rank, where $vec(\cdot)$ denotes the vectorization operation of a matrix, i.e., cascading the column vectors of a matrix. Matrix $O \in R^{M^{2N} \times N^N}$ with each element

$$o_{M^N(i-1)+j, \sum_{r=1}^{N}(m_r-1)N^{N-r}+1} = \prod_{r=1}^{N} a_{i_{m_r}, j_r}, \tag{9}$$

where

$$\begin{aligned} &1 \le i, j \le M^N, \\ &1 \le m_1, \cdots, m_N \le N, \end{aligned}$$

and i_{m_r} denotes the weather status of region m_r which is consistent with the area's weather state i of G. $V \in R^{N^2 \times N^N}$ with each element

$$v_{N(l-1)+n, \sum_{r=1}^{N}(m_r-1)N^{N-r}+1} = \begin{cases} 1, & if\ m_l = n \\ 0, & otherwise \end{cases}, \tag{10}$$

where

$$1 \leq l, n \leq N.$$

The reverse mapping from G to D can be derived by multiplying (7) out and construct two matrices O and V for calculation. Note that matrix O is solely dependent on A, which can be extracted from G directly as shown in *a)*. Matrix V is solely dependent on N.

c). Given a master Markov chain G constructed from a network influence matrix D and a local Markov chain A, if $Null(O) \supset Null(V)$, D can not be uniquely determined from G. In other words, multiple different matrices Ds can map to a same G. In this case, the influence model is not identifiable from G, and hence not identifiable from spatiotemporal weather scenarios. Since the estimation is meaningful only when there is a one-to-one mapping between the scenario and the model parameters, we here focus on the cases where A and D can be uniquely determined from the scenarios. Matrices O and V can be used to check the design of the spatiotemporal weather spread scenarios generator.

C. Estimation Algorithms for Influence Model

In this subsection, we propose two estimation algorithms for the influence model driven spatiotemporal scenarios. The first algorithm is the widely used MLE, and the second algorithm called linear-algebra-based influence model estimator is derived based on the unique properties of the influence model.

1. Maximum Likelihood Estimation Algorithm

Given a spatiotemporal weather scenario Y_i which is generated from an influence model, our goal is to find the underlying parameter $\theta_i = (A_i, D_i)$ that maximizes the likelihood function $L(\theta_i; Y_i)$.

$$L(\theta_i; Y_i) = P(Y_i|\theta_i), \tag{11}$$

$$\hat{\theta}_i = \arg\max_{\theta_i} L(\theta_i; Y_i). \tag{12}$$

After mapping the influence model to the master Markov chain G, we have $Y_i = \left[s^i[1], s^i[2], \cdots, s^i[|T|]\right]$, where $s^i[k]$ denotes the weather state of the area corresponding to scenario Y_i at time k, and $|T|$ is the temporal length as indicated in Section III. Equation (11) can be expressed as:

$$L(\theta_i; Y_i) = P(Y_i|\theta_i) = P(s^i[1], s^i[2], \cdots, s^i[|T|]|\theta_i) = \prod_{k=1}^{|T|-1} P(s^i[k+1]|s^i[k], \theta_i) \tag{13}$$

To facilitate computation, take log of (13), we have

$$\begin{aligned} &logL(\theta_i; Y_i) = \sum_{k=1}^{|T|-1} logP(s^i[k+1]|s^i[k], \theta_i), \\ &s.t. \sum_{n=1}^{M} a_{mn} = 1, \ \sum_{r=1}^{N} d_{lr} = 1, \\ &1 \leq m \leq M, 1 \leq l \leq N. \end{aligned} \tag{14}$$

By applying the Lagrange multiplier to (14), $\hat{\theta}_i$ is obtained.

2. Linear-Algebra Based Estimation algorithm

Given a scenario Y_i which is generated from an influence model, the corresponding master Markov chain G_i can be obtained directly by counting the state transition frequencies. As the time approaches infinity, G_i will be equal to its real value with probability 1 according to the law of large numbers. According to the mapping relationship from G_i to the local Markov chain A_i and the network influence matrix D_i, the linear-algebra based estimation algorithm can be described as follows.

Algorithm 2 Linear-Algebra Based Influence Model Estimator

Input:

A spatiotemporal weather scenario Y_i.

Output:

Matrices A_i and D_i underlying Y_i.

1: Count state transition frequencies according to Y_i, then calculate matrix G_i.
2: Compute the local Markov chain A_i from G_i according to (8).
3: Compute matrix O according to (9).
4: **if** matrix O has full column rank **then**
Compute network influence matrix D_i based on G_i, O and V according to $vec(D^T) = V(O^T O)^{-1} O^T vec(G^T)$.
5: **end if**

D. Three Model-Based Distance Measures

A number of model-based distance measures have been developed for HMMs. A commonly used framework is to obtain a likelihood-based distance matrix for all the scenarios based on KL divergence. Considering the mapping relationship between the influence model and HMM, the model-based distance measures for HMM can be transplanted to influence model naturally.

An earlier approach for measuring the distance between pairwise HMMs was introduced in [11]. The approach followed the concept of divergence and cross entropy in information theory and developed a probabilistic distance measure. Given a scenario Y_i, the distance between a pair of models θ_1 and θ_2 is defined as:

$$l_{12} = \frac{1}{|T|}(\log P(Y_i|\theta_1) - \log P(Y_i|\theta_2)), \tag{15}$$

where l_{12} measures the difference of the likelihood of generating the same scenario from two different models. To extend the results to multiple scenarios, for example $Y = \left[Y_1, Y_2, \cdots, Y_L\right]$, where L denotes the number of scenarios, the main idea is to estimate a model for each scenario and use the resulting models to calculate a length-normalized log-likelihood matrix W with each element

$$w_{ij} = \frac{1}{|T|} \log P(Y_j|\hat{\theta}_i), \tag{16}$$

where w_{ij} expresses the likelihood of generating scenario Y_j using the underlying model $\hat{\theta}_i$. Since $\hat{\theta}_i$ is estimated from Y_i, the more similar Y_i and Y_j are, the larger w_{ij} is. In the following, we introduce three model-based distance measures based on matrix W for clustering the influence model driven spatiotemporal weather scenarios.

1. Simple Symmetric Distance Measure

The simple symmetric distance measure is the simplest way to obtain a distance matrix $\mathcal{D}^{SS}$. Each entry $\mathcal{D}^{SS}_{i,j}$ of $\mathcal{D}^{SS}$ is calculated by taking half of the summation of w_{ij} and w_{ji}. That is,

$$\mathcal{D}^{SS}_{i,j} = \mathcal{D}^{SS}_{j,i} = \frac{1}{2}(w_{ij} + w_{ji}) \tag{17}$$

(17) incorporates the information of both θ_i and θ_j and the two scenarios Y_i and Y_j, and guarantees $\mathcal{D}^{SS}$ to be symmetric.

2. BP Metric

In [12], a model-based distance measure named BP metric is proposed. Each entry $\mathcal{D}^{BP}_{i,j}$ of $\mathcal{D}^{BP}$ is calculated as

$$\mathcal{D}^{BP}_{i,j} = \frac{1}{2}(\frac{w_{ij} - w_{ii}}{w_{ii}} + \frac{w_{ji} - w_{jj}}{w_{jj}}) \tag{18}$$

Compared with (17), (18) takes into account the performance of the estimator. The simple symmetric distance measure assumes that all the scenarios are estimated with the same accuracy. However, due to different spatiotemporal

spread patterns, the performances of the estimators for the scenarios may be different, even though they use the same estimation algorithm. The information of how well the scenarios are estimated by the estimator is expressed in w_{ii} and w_{jj}, by involving them into (18), the BP metric is expected to perform better than the simple symmetric distance measure when the performance of estimator differs for different model structures.

3. Yin-Yang distance

In [13] another distance measure named Yin-Yang distance is proposed as follows.

$$\mathcal{D}_{i,j}^{YY} = |w_{ii} + w_{jj} - w_{ij} - w_{ji}| \tag{19}$$

Like BP metric, Yin-Yang distance also takes into account how well the scenarios are estimated by the estimator. If the two scenarios are identical, $\mathcal{D}_{i,j}^{YY}$ would be equal to zero. If the two scenarios differ greatly from each other, their likelihood of being generated from the other model would be small and the distance between them would be large. It is applicable to long-length scenarios with large noises.

E. Hierarchical Clustering Algorithm

In this subsection, we briefly introduce the hierarchical clustering algorithm based on the distance matrix as follows.

Algorithm 3 Hierarchical Clustering Algorithm

Input:

Scenarios $Y = [Y_1, Y_2, \cdots, Y_L]$ and a distance matrix $\mathcal{D}$.

Output:

Clustering results.

1: Assign each scenario to a cluster. In particular,we have L clusters and each of them contains only one scenario.
2: Find the closest (the minimum distance) pair of clusters based on $\mathcal{D}$ and merge them into a new cluster.
3: Calculate the distances between the new cluster and each of the old clusters using single-linkage, i.e., the distance between one cluster and another cluster is calculated as the shortest distance from any member of one cluster to any member of the other cluster. Update the distance matrix $\mathcal{D}$ accordingly.
4: Repeat steps 2 and 3 until all scenarios are clustered into a single cluster of size L.

F. The Framework For Influence Model-based Clustering Algorithm

Combining the influence model estimation algorithms, the three model-based distance measures, and the hierarchical clustering algorithm, the framework for influence model-based clustering algorithm for spatiotemporal weather scenarios are summarized as follows.

Step 1: Given multiple spatiotemporal weather scenarios $Y = [Y_1, Y_2, \cdots, Y_L]$, estimate the influence model parameters underlying each scenario according to Algorithm 2.

Step 2: Apply the model-based distance measures aforementioned in Subsection IV.D to the model parameters acquired in *Step 1* and obtain a distance matrix $\mathcal{D}$ for Y.

Step 3: Apply the hierarchical clustering algorithm described in Algorithm 3 to the distance matrix $\mathcal{D}$, the clustering results of the spatiotemporal weather scenarios Y can be obtained.

V. Comparative Simulation Studies

A. Spatiotemporal Weather Spread

We generate 100 spatiotemporal weather spread scenarios of four different spread patterns as shown in Figure 2. We model how a cold front intrudes an area which is consist of 20×20 regions. The black dot denotes a region affected by he cold front, and the white one denotes a region of normal weather. We use the binary influence models to generate the four weather spreading patterns by choosing an appropriate network influence matrix D to ensure our desired propagation direction.

Then we apply the three model-based distance measure mentioned in Section IV and the data-driven multi-resolution distance measure mentioned in Section III to cluster the 100 scenarios. Their performances are shown in Table 1.

Table 1 Accuracy of The Distance Measures For Clustering Spatiotemporal Weather Scenarios

Accuracy (Percent)			
Simple Symmetric Distance	BP Metric	Yin-Yang Distance	Data-based Clustering Algorithm
75	74	75	98

The data-driven multi-resolution distance shows better performance than the three model-based approaches in this case. Since the four weather spreading patterns are generated by binary influence models and binary influence models are mapped to absorbing Markov chains which are not ergodic, the model-based approach needs large data sets to be accurate. With small datasets, the model-based approaches lead to less accuracy. For the three model-based approaches, the accuracy of them differ little from each other in this case.

B. Scenario Length Study

Here we study the relationship between scenario length and the performance of the clustering results of the model-based approaches and the data-driven approach. To reduce computational complexity, we adopt four influence models with a similar graph structure as shown in Figure 1. All the four influence models have different network influence matrices D and local Markov chains A. Scenario length ranges from 100 to 2000, under each case 100 scenarios are generated by these four models accordingly. The three model-based approaches and the data-driven approach are applied to cluster the 100 scenarios under each length respectively. The accuracy of these approaches under each length is given in Figure 3. With the increasing of scenario length, all of the model-based approaches and the data-driven approach improve their accuracy. There is no significant differences in the performance of the three model-based approaches. However, the performance of the multi-resolution distance approach is much inferior to the model-based approaches in this case. Since the spatiotemporal weather spreading patterns are not as evident as in V.A, it is difficult to select appropriate parameters for the multi-resolution distance approach to cluster the spreading patterns from the scenarios.

C. Study on Scenario Data of Heterogeneous Lengths

We study the clustering of heterogeneous-length scenarios for the three model-based distance approaches since the data-driven multi-resolution approach requires all the scenarios to have the same length. We adopt the same four influence models as in case study V.B, and generate 100 scenarios from them accordingly, of which number 1-25 scenarios are generated by the first influence model, number 26-50 scenarios are generated by the second influence model, number 51-75 scenarios are generated by the third influence model, and number 76-100 scenarios are generated by the forth influence model. The lengths of these scenarios are not the same and are randomly determined within a range from 100 to 2000. The accuracy of the three model-based approaches is shown as in Table 2. We can see BP

Table 2 Accuracy of Distance Measures Against Varied-length Scenarios

Accuracy (Percent)		
Simple Symmetric Distance	BP Metric	Yin-Yang Distance
71	78	79

metric and Yin-Yang distance outperform the simple symmetric distance in clustering heterogeneous-length scenarios. That is because BP metric and Yin-Yang distance take into account how well the estimator works and evaluates the relative difference. There is no prominent difference between the performance of BP metric and Yin-Yang distance in this case.

VI. Conclusion

In this paper, we develop an influence-model based clustering algorithm to cluster spatiotemporal weather scenarios for air traffic management. We use the influence model to capture the dynamics of stochastic spatiotemporal weather

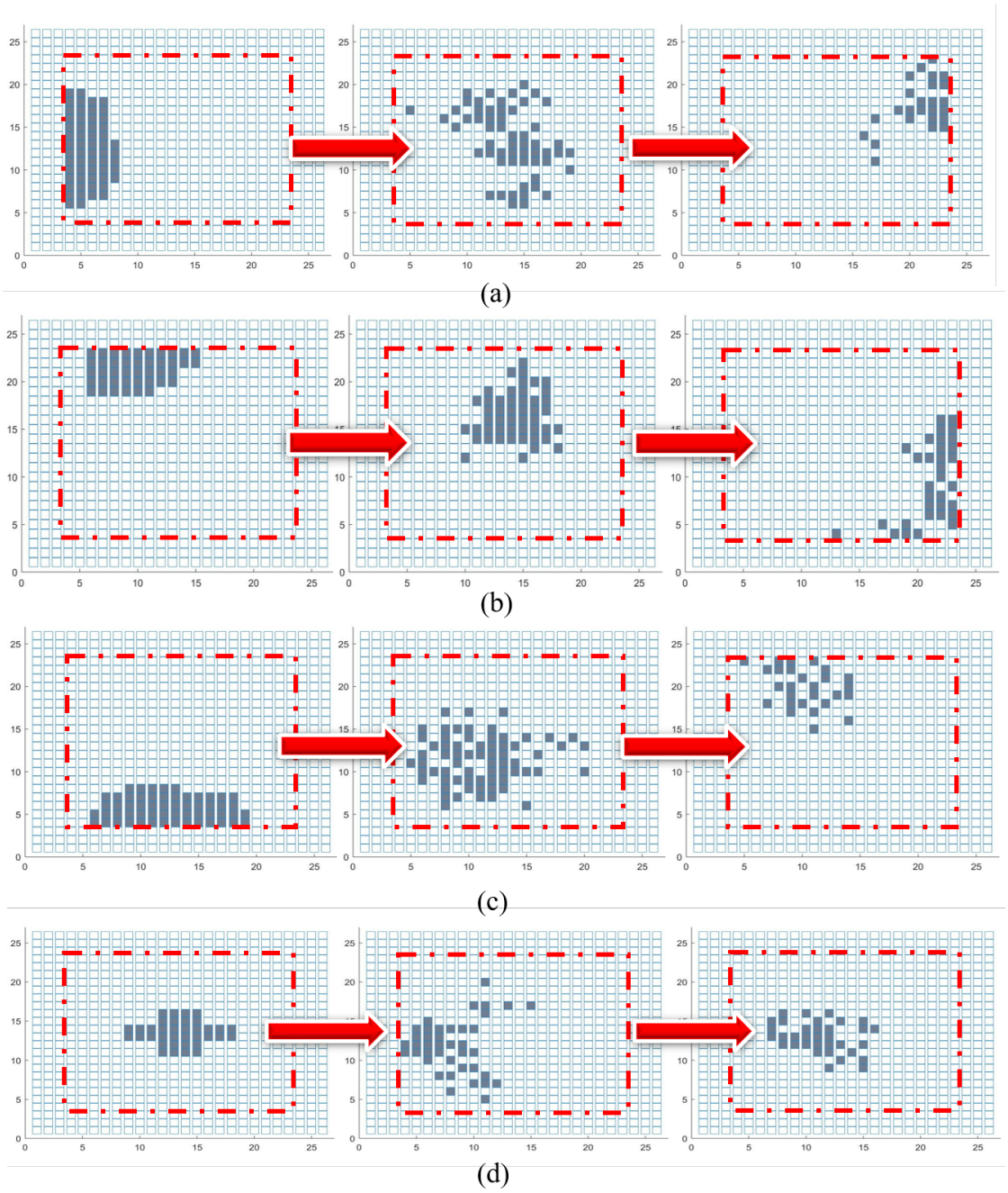

Fig. 2 Four spatiotemporal weather spread patterns. (a) The cold front comes from the west and spreads to the east.(b) The cold front comes from the northwest and spreads to the southeast. (c) The cold front comes from the south and spreads to the north. (d) The cold front starts in the middle and spreads to the regions around.

scenarios, and study the estimation algorithms for the influence model based on its unique properties. Three influence model-based distance measures for clustering stochastic weather scenarios are studied and compared with a data-driven multi-resolution spatiotemporal distance measure. Combining the influence model estimator, three model-based distance measures, and the hierarchical clustering algorithm, we develop a influence model-based clustering algorithm for spatiotemporal weather scenarios. The simulation studies show that the performance of the three model-based distance measures are more robust and require less user interference than the data-driven multi-resolution spatiotemporal distance measure, which can be easily affected by the selected parameters. With the increase of scenario length, the clustering

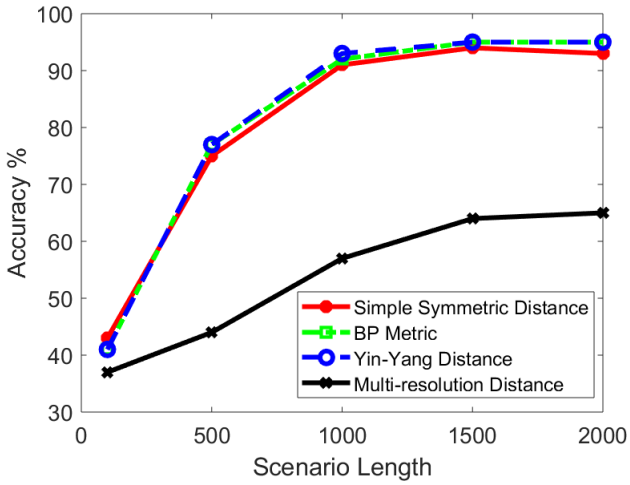

Fig. 3 Accuracy Under Different Scenario Length

accuracy of the model-based distance measures increase substantially. In addition, the model-based distance measures are applicable to the clustering of scenarios with different lengths.

Acknowledgments

We acknowledge the National Science Foundation under Grant 1839707, 1724248 and 1714826 for the support of this work.

References

[1] Zhou, Y., Wan, Y., Roy, S., Taylor, C., Wanke, C., Ramamurthy, D., and Xie, J., "Multivariate probabilistic collocation method for effective uncertainty evaluation with application to air traffic flow management," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 44, No. 10, 2014, pp. 1347–1363.

[2] Xie, J., Wan, Y., and Lewis, F. L., "Strategic air traffic flow management under uncertainties using scalable sampling-based dynamic programming and Q-learning approaches," *2017 11th Asian Control Conference (ASCC)*, IEEE, Gold Coast, Australia, 2017, pp. 1116–1121.

[3] Wan, Y., Taylor, C., Roy, S., Wanke, C., and Zhou, Y., "Dynamic queuing network model for flow contingency management," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 14, No. 3, 2013, pp. 1380–1392.

[4] Taylor, C., Wanke, C., Wan, Y., and Roy, S., "A framework for flow contingency management," *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, including the AIAA Balloon Systems Conference and 19th AIAA Lighter-Than*, Virginia Beach, VA, 2011, p. 6904.

[5] Xie, J., Reddy Kothapally, A., Wan, Y., He, C., Taylor, C., Wanke, C., and Steiner, M., "Similarity Search of Spatiotemporal Scenario Data for Strategic Air Traffic Management," *Journal of Aerospace Information Systems*, 2019, pp. 1–16.

[6] He, C., Wan, Y., and Xie, J., "Spatiotemporal scenario data-driven decision for the path planning of multiple UASs," *Proceedings of the Fourth Workshop on International Science of Smart City Operations and Platforms Engineering, SCOPE@CPSIoTWeek*, Montreal, Canada, 2019.

[7] Xie, J., Wan, Y., Zhou, Y., Tien, S.-L., Vargo, E. P., Taylor, C., and Wanke, C., "Distance Measure to Cluster Spatiotemporal Scenarios for Strategic Air Traffic Management," *Journal of Aerospace Information Systems*, Vol. 12, No. 8, 2015, pp. 545–563.

[8] Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A. Y., Foufou, S., and Bouras, A., "A survey of clustering algorithms for big data: Taxonomy and empirical analysis," *IEEE transactions on emerging topics in computing*, Vol. 2, No. 3, 2014, pp. 267–279.

[9] Asavathiratham, C., "The influence model: A tractable representation for the dynamics of networked markov chains," Ph.D. thesis, Massachusetts Institute of Technology, 2001.

[10] García-García, D., Hernández, E. P., and Díaz-de María, F., "A new distance measure for model-based sequence clustering," *IEEE transactions on pattern analysis and machine intelligence*, Vol. 31, No. 7, 2009, pp. 1325–1331.

[11] Juang, B.-H., and Rabiner, L. R., "A probabilistic distance measure for hidden Markov models," *AT&T technical journal*, Vol. 64, No. 2, 1985, pp. 391–408.

[12] Panuccio, A., Bicego, M., and Murino, V., "A Hidden Markov Model-based approach to sequential data clustering," *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, Springer, Windsor, Canada, 2002, pp. 734–743.

[13] Yin, J., and Yang, Q., "Integrating hidden Markov models and spectral analysis for sensory time series clustering," *Data Mining, Fifth IEEE International Conference on*, IEEE, Washington, DC, 2005, pp. 8–pp.

[14] He, C., Wan, Y., and Lewis, F. L., "On the Identifiability of the Influence Model for Stochastic Spatiotemporal Spread Processes," *2019 American Control Conference*, Philadelphia, USA, 2019.