Modeling the Effect of Urbanization on Climate and Dust Generation Over Desert Cities

by

Sherzad Tahseen Tahir

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2019 by the
Graduate Supervisory Committee:

Huei-Ping Huang, Chair
Patrick Phelan
Marcus Herrmann
Kangping Chen
Amanda Clarke

ARIZONA STATE UNIVERSITY

May 2019

ABSTRACT

Understanding and predicting climate changes at the urban scale have been an important yet challenging problem in environmental engineering. The lack of reliable long-term observations at the urban scale makes it difficult to even assess past climate changes. Numerical modeling plays an important role in filling the gap of observation and predicting future changes. Numerical studies on the climatic effect of desert urbanization have focused on basic meteorological fields such as temperature and wind. For desert cities, urban expansion can lead to substantial changes in the local production of wind-blown dust, which have implications for air quality and public health. This study expands the existing framework of numerical simulation for desert urbanization to include the computation of dust generation related to urban land-use changes. This is accomplished by connecting a suite of numerical models, including a meso-scale meteorological model, a land-surface model, an urban canopy model, and a turbulence model, to produce the key parameters that control the surface fluxes of wind-blown dust. Those models generate the near-surface turbulence intensity, soil moisture, and land-surface properties, which are used to determine the dust fluxes from a set of laboratory-based empirical formulas. This framework is applied to a series of simulations for the desert city of Erbil across a period of rapid urbanization. The changes in surface dust fluxes associated with urbanization are quantified. An analysis of the model output further reveals the dependence of surface dust fluxes on local meteorological conditions. Future applications of the models to environmental prediction are discussed.

DEDICATION

To my parents and best teachers for their time and efforts in helping me attain this achievement. To my spouse Hadar and my siblings for their unceasing support and encouragement through my work on this research project. To everyone who has taught me and helped enrich my knowledge, all the way for me to reach the level I am at now. Without you, I would not have been able to complete this dissertation.

ACKNOWLEDGMENTS

I would like to acknowledge Dr. Huei-Ping Huang for his great support and prudent guidance through every step of this work. Without Dr. Huang's prodigious mentoring techniques, the completion of this dissertation would not have been possible.

I would like to express my grate gratitude to the committee members, Dr. Patrick Phelan, Dr. Marcus Herrmann, Dr. Kangping Chen, and Dr. Amanda Clarke for their continuous support and suggestions that improved this work. I also thank Kyle Mohr for useful discussions on dust transport.

I thank Dr. Marc Mignolet, Chair of MAE Graduate Program, and staff members of MAE Graduate Advising Office for valuable administrative support during the course of this study. I gratefully acknowledge the support by an Engineering Graduate Fellowship in 2018, a Completion Fellowship of ASU Graduate College in 2019, and additional support provided by the College of Engineering at the University of Duhok and by Arizona Nutritional Supplement, which helped make the timely completion of this work possible.

TABLE OF CONTENTS

Page

CHAPTER Page

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

CHAPTER 1

INTRODUCTION

1.1 Background

The rapid urban expansion that occurred worldwide in the last decades has profound impacts on global environment and sustainability. In particular, urban development and population growth have occurred very rapidly over arid and semiarid regions (United Nations 2017), despite the fact that those regions are already under high environmental stresses. It is a major challenge in both basic science and applications to quantify the effect of urbanization on local weather, climate, and air quality. Urbanization is a process of a conversion of natural or agricultural lands to concrete and impervious surfaces, as exemplified by the summary of recent land-use changes for the United States as shown in Fig. 1. The changes in land surfaces imply physical changes in heat capacity, surface friction, soil moisture, and the composition of soil. These changes could potentially lead to changes in regional meteorological fields such as wind, temperature, and precipitation (Rasul, 2015, Kamal et al. 2015b, 2017). For desert cities, a unique aspect is the potential change in dust generation associated with urbanization. Because dust generation depends on the strength of surface wind and soil moisture, the process is coupled to the regional scale meteorological fields.

With few observations, quantification of these processes relies mainly on numerical simulations using advanced computer models. Previous studies on regional climate changes related to desert urbanization have focused on the meso-scale meteorological fields (e.g., Kamal *et al*. 2015b, 2017) but rarely connected them to dust production. This study will use a suite of numerical models to fill this gap of knowledge, improving the framework for the assessment and prediction of not only meteorological fields but also surface dust fluxes.

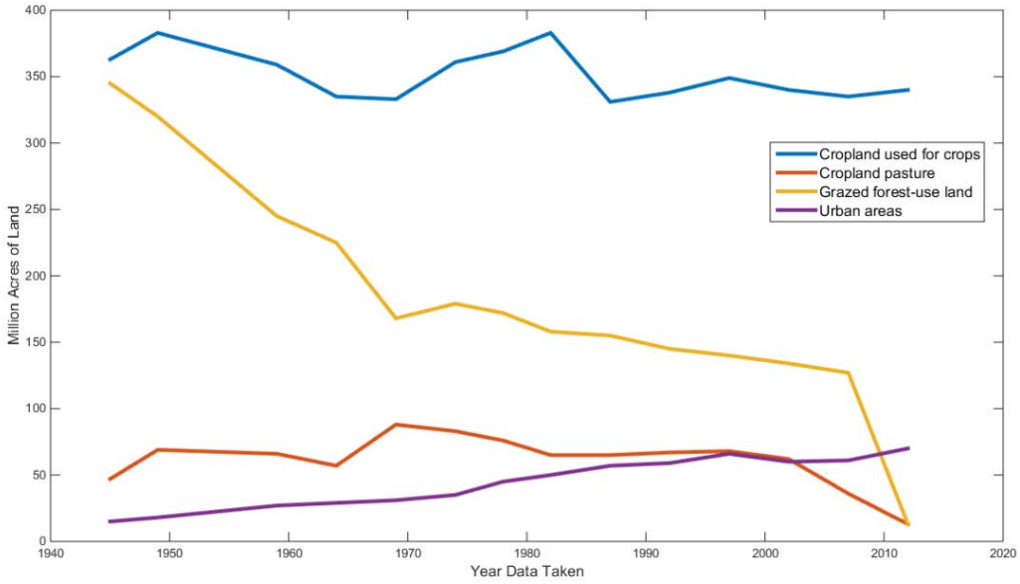

Figure 1: Examples of the areas of some major Land Use categories in the United States as they evolve from1945 to 2013. The areas of Cropland (blue), Cropland with pasture (brown), and Grazed forest (Yellow) have declined while the area of Urban (purple) region has increased.

To realize the goals of this study, we will use a suite of numerical models. The tools needed to compute dust generation include: (i) An environmental fluid dynamics model that helps simulate the large scale meteorological fields, (ii) A land surface model that computes the interaction between the land surface and the atmosphere, (iii) A turbulence model that conveys the information of large scale meteorological fields into

the intensity of near surface turbulence, and (iv) An empirical model that translates near surface turbulence intensity and soil moisture into dust fluxes. These essential components have been developed individually by previous studies as detailed in Chapter 2. However, it is highly nontrivial to seamlessly connect all of them together. The main technical ingredient of this study is in facilitating the series of connection of numerical models in a realistic setting, using the observed scenario of urbanization of a rapidly developing desert city as the backdrop to complete the numerical simulations.

The complete framework from dynamical downscaling for meteorological fields to the estimation of surface dust fluxes is described in Chapter 2. This framework is tested on a rapidly growing desert city, Erbil in Northern Iraq. The incorporation of land-use changes of the city into the numerical models is described in Chapter 3. The validation and verification of the models are carried out in Chapter 4 and 5, and the key results on dust generation are presented in Chapter 6.

In general, airborne dust can be produced by natural and anthropogenic processes. Due to a general lack of pollution inventory for industrial dust generation over desert cities, this study considers only natural wind-blown dust. The numerical simulations quantify the amount of wind-blown dust generated in “suburbia” in the pre-urbanization era, which is largely suppressed after urban expansion when desert-type of land surfaces are replaced by concrete. The results fill a gap in observation for the past. Moreover, from the model output, the relation between dust generation and local meteorological conditions is quantified.

We discuss future work on dust transport and model improvements in Chapter 7. Chapter 8 summarizes the key achievement of this study and provides recommendations for applications.

CHAPTER 2

NUMERICAL MODELS AND METHODOLOGY

2.1 Overview

To incorporate the required multiple tools described in Chapter 1 into a unified framework for our numerical simulations, we choose to use the Weather Research and Forecasting (WRF) model as the starting point. The WRF model is an open source computational fluid dynamics solver developed for simulating large-scale environmental flows (Skamarock, et al., 2008); its latest version has been used in practical daily weather forecasts. With a large computational domain for typical environmental applications, it is not possible to run the model at a very high resolution. Also, environmental fluid systems are strongly forced by diabatic heating from solar radiation and latent heat release by precipitation. Large-scale flows are generally in the flow regime with a very high Reynolds number; the influence of turbulence is expected to be strong yet true turbulent motion cannot be resolved by the model. The WRF model addresses these issues by coupling the "dynamical core" for Navier-Stokes equations to many modules of physical and subgrid-scale parameterization. This feature is highly relevant to our study, because the estimate of dust fluxes requires the information of near-surface turbulence intensity and soil moisture (influenced by rainfall) which will come from those parameterization schemes.

For our purpose, it is also relevant that the WRF model is already equipped with a land-surface model which allows detailed calculations of the exchange of heat, moisture, and momentum fluxes between the atmosphere and land surface. This feature was critical because those fluxes change significantly with a change of land cover associated with urbanization. The land model also facilitates dynamic predictions of soil moisture across multiple underground layers. The soil moisture at the surface is a key parameter for computing dust fluxes. What remains to be determined is the relation that connects the WRF output of meteorological and surface variables to dust fluxes. This part of the computation is done outside the WRF package but instead relies on empirical relations from laboratory experiments.

2.2 The Dynamical Core of WRF Model

The dynamical core of WRF is based on finite-difference C-grid for spatial discretization, with staggered grids in both horizontal and vertical directions (Fig. 2). The vertical coordinate is terrain-following (Fig. 3). This allows an easy reconfiguration of the grid system over an area with complex topography. As a trade-off, the vertical coordinate depends on surface pressure (or vertically integrated mass of the atmospheric column) which needs to be predicted. More precisely, the vertical coordinate is defined by:

$$\eta = (P_h - P_{ht})/\mu, \qquad (1)$$

where $\mu = (P_{hs} - P_{ht})$, P_h is the hydrostatic pressure, and P_{hs} and P_{ht} are the pressure at the surface and the top of atmosphere.

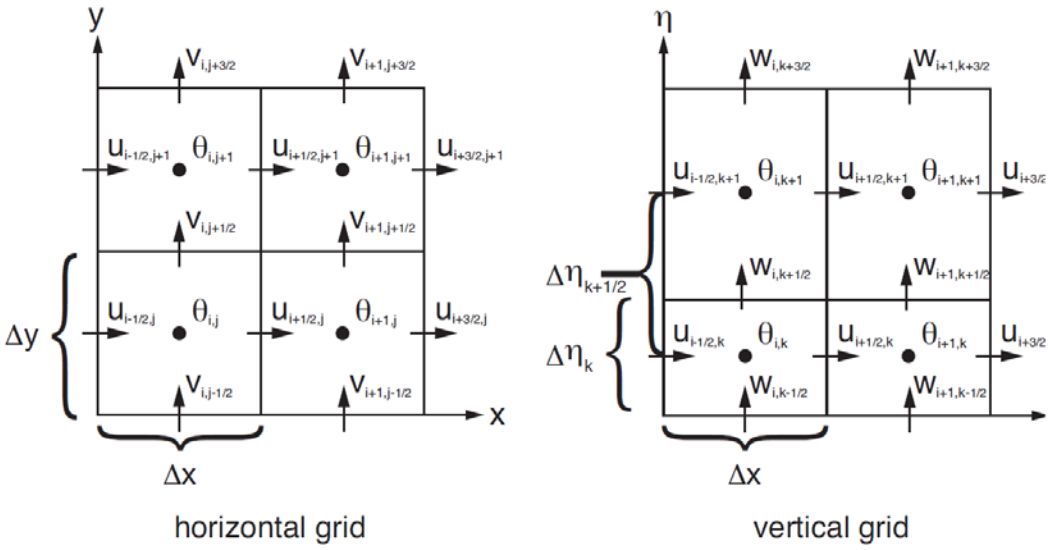

Figure 2: The horizontal and vertical grid configurations in WRF (Skamarock, et al., 2008)

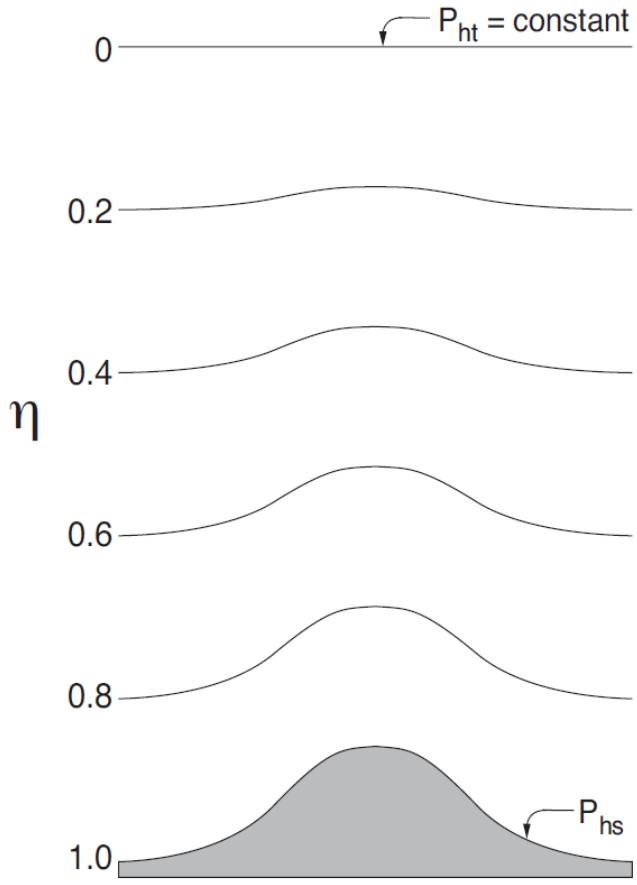

Figure 3: The terrain-following vertical coordinate (η) in WRF (Skamarock, et al., 2008)

The governing equations for the dynamical core are (Skamarock, et al., 2008):

$$\partial_t U + (\nabla \cdot \mathbf{V}u) + \mu_d \alpha \partial_x p + (\alpha/\alpha_d)\partial_\eta p \partial_x \varphi = F_U \quad (2)$$

$$\partial_t V + (\nabla \cdot \mathbf{V}v) + \mu_d \alpha \partial_y p + (\alpha/\alpha_d)\partial_\eta p \partial_y \varphi = F_V \quad (3)$$

$$\partial_t W + (\nabla \cdot \mathbf{V}w) - g[(\alpha/\alpha_d)\partial_\eta p - \mu_d] = F_W \quad (4)$$

$$\partial_t \Theta + (\nabla \cdot \mathbf{V}\theta) = F_\Theta \quad (5)$$

$$\partial_t \mu_d + (\nabla \cdot \mathbf{V}) = 0 \quad (6)$$

$$\partial_t \varphi + \mu_d^{-1} [(\mathbf{V} \cdot \nabla \varphi) - gW] = 0 \quad (7)$$

$$\partial_t Q_m + (\nabla \cdot \mathbf{V}q_m) = F_{Qm} \quad (8)$$

$$\partial_\eta \varphi = -\alpha\mu \quad (9)$$

$$p = p_0 (R_d \theta / p_0 \alpha)^\gamma \quad (10)$$

Equations (2)-(4) are the momentum equations, (5) is thermodynamic energy equation, and (6) and (7) are prognostic equations for mass. (The need to have two, instead of one, equations for mass continuity is due to the use of the η coordinate). In those equations, $\boldsymbol{V} = (u, v, w)$ are the 3-D velocities, p is pressure, g is gravity, $\phi = gz$ is geopotential, $\theta = T(p/p_0)^{-R/Cp}$ is potential temperature (where T is temperature, p_0 is a reference pressure and R and Cp are the ideal gas constant and the heat capacity of the atmosphere). The α and α_d in Eqs. (2)-(4) are the inverses of total and dry-air density, respectively. They are related by $\alpha = \alpha_d(1+q_v+q_c+q_r+q_i+\ldots)^{-1}$,where q is the mixing ratio and the subscripts denote various forms of water such as water vapor, liquid water in cloud and rain, and solid water in ice crystals in clouds. The transport equations for those moist variables are Eq. (8) with different subscripts of q. The μ_d in Eqs. (4), (6), and (7) is the vertically integrated mass of dry air (in contrast to the μ introduced in Eq. (1) which is the vertically integrated total mass including contributions from various phases of water). To

accommodate the use of terrain-following vertical coordinate, the actual prognostic variables are $(U, V, W, \Theta) = \mu\,(u, v, w, \theta)$, and $Q_m = \mu_d\, q_m$ for the moist variables. Equation (9) is a diagnostic equation for ϕ and α based on the hydrostatic relation. (Note that the model as a whole is non-hydrostatic.) Lastly, Eq. (10) is the (ideal gas) equation of state for the atmosphere.

The Coriolis force in full spherical coordinates is included in the F_U and F_V in the right hand side of Eqs. (2) and (3). Its detailed form is omitted for brevity. However, it is important to mention that the model is constructed under the rotating frame (i.e., the velocity in the model is relative to the rotating Earth). The presence of the Coriolis force in the momentum equation serves to represent the effect of Earth rotation.

It is equally important to note that the F_U, F_V, F_W, and $F\theta$ in Eqs. (2)-(5) collect all the diabatic forcing coming from the added modules for physical and subgrid-scale parameterization. For a typical simulation of large-scale environmental flows, they are as important as the bare-bone "dry dynamics" described in the left hand side of the equations. For example, $F\theta$ includes the heating by solar radiation and by condensation associated to moist convection. A turbulence model embedded in WRF would generate momentum source or sink for the resolved flow by turbulence momentum transport, which is incorporated into F_U, F_V and F_W in the momentum equation.

As is obvious from Eq. (4), the model is non-hydrostatic in the vertical direction (such that vertical velocity is a full prognostic variable). This allows a more accurate representation of thermal convection but it permits acoustic modes in the model, which complicates the design of the numerical scheme in time. The time integration uses a split scheme: Within a long step that uses the third-order Runge-Kutta (RK3) scheme, acoustic

integration is performed in many shorter steps in a quasi-linear manner (i.e., with the non-acoustic large-scale component “frozen”). The advective Courant number will bound the RK3 time step; WRF allows users to impose the step size or let the model self-adjust according to the stability criterion. The typical RK3 time step size used in the long simulations (each over a month or a season) in this study is around 3 minutes. Further details for the setup of WRF are in Appendix A.

2.3 Boundary Conditions and Nesting

In applications, WRF has been used more extensively for short-term weather predictions. For our purpose of climate modeling, long simulations that last a season (or longer) are required. (In particular, the effect of land-use change on the weather would be masked out by short-term noise but the signal becomes clear after seasonal or long-term average). For a regional model, to prevent a “climate drift”, such long simulations need to be constrained by realistic lateral boundary conditions. A commonly used strategy adopted in this study is broadly named “dynamical downscaling”, in which the time-varying large-scale meteorological variables (from observation or global climate model simulations) were imposed at the lateral boundary of WRF model domain. Multiple layers of nesting of the WRF grids were employed to allow a high-resolution (by the standard of environmental fluid simulation) run over the “target” location. In this manner, WRF generated the high-resolution 3-D meteorological fields over the highly resolved inner domain where no observation of comparable resolution was available.

To facilitate dynamical downscaling for the long simulations, we used the global 4-times daily observational data (from NOAA FNL data set, NCAR and UCAR Research Data Archive) to constrain the lateral boundary condition for the outermost WRF model domain. The global data set has only an approximately a 1-degree (latitude/longitude), or ~ 100 km, horizontal resolution. To perform a high-resolution simulation over an urban area, the desired resolution is close to 1 km. This requires multiple layers of nesting as described below.

The WRF modeling system allows multiple layers of nesting, in the configuration illustrated by (a) or (b) in Fig.4. For our study, as we focus on a single city, configuration (a) was chosen. With the aforementioned staggered grids in WRF, nesting requires specific rules as illustrated in Fig. 5: The ratio of the grid sizes between two layers of high- and low-resolution grids must be an integer.

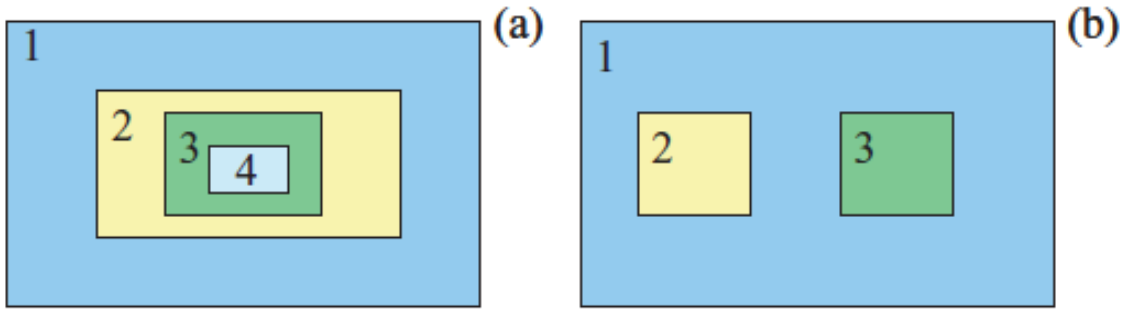

Figure 4: Nesting configuration for multiple grids (a) Telescoping nests (b) Same level nesting with respect to parent grid (Skamarock, et al., 2008)

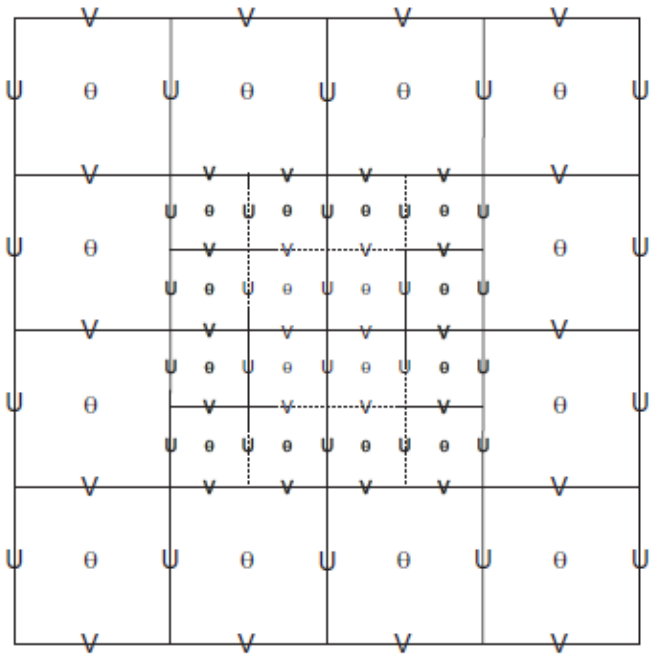

Figure 5: A schematic diagram of nesting of staggered grids. Shown are the parent domain and an imbedded nested domain with a 3:1 grid size ratio. U and V are the horizontal velocity components and θ is the thermodynamic variable (Skamarock, et al., 2008).

The WRF package comes with a high-resolution geographical information dataset. With nesting into ever increasing resolution, a higher resolution version of the surface topography and land cover type is automatically generated. Further details on the selection of model domains and setups for nesting are postponed to Chapter 3.

2.4 Execution of Climate Simulations Using WRF

The dynamical core of WRF is coupled to many subroutines for the additional physical and subgrid-scale processes. This package, in turn, is supported by more subroutines for external input and output, as summarized in Fig. 6. In particular, the system includes a set of tools for initialization. For this purpose, we choose the 3-D VAR scheme and use the global FNL data as the starting point for preprocessing of the initial

condition. We also adjusted the output by retaining the variables relevant to this study. As noted in Section 2.3, for a long "climate" simulation we constrained the lateral boundary conditions (imposed at the boundary of the outermost domain) with time-varying large-scale meteorological fields from observation. In all of our simulations, the lateral boundary conditions came from the 6-hourly FNL data as described before.

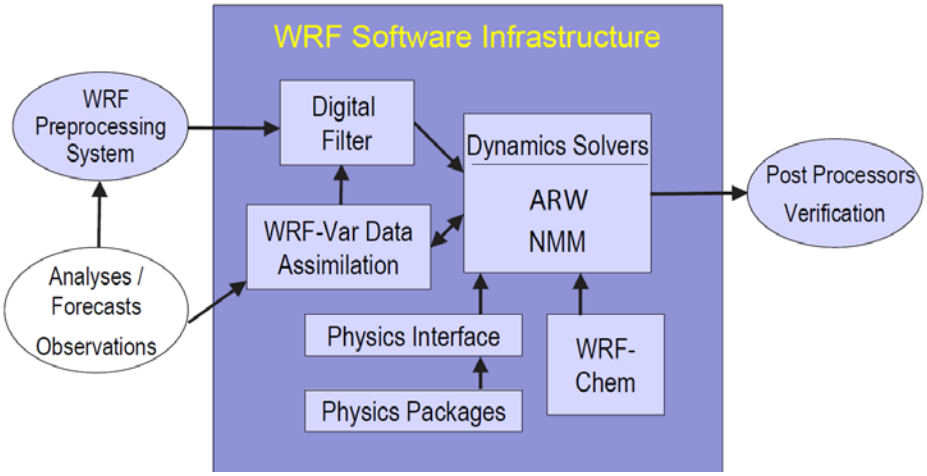

Figure 6: The main structure of the WRF System (Skamarock, et al., 2008).

2.5 Turbulence Model

The WRF system described in the preceding sections produced the large-scale meteorological fields at the model grid, which was still relatively coarse for our application. The model did not explicitly resolve turbulence in the planetary boundary layer which is relevant to dust generation. To fill this missing link, we activated a turbulence model embedded in WRF to interactively to calculate relevant parameters such as the eddy momentum and heat fluxes due to subgrid-scale turbulence, but particularly the friction velocity, u^*, which was later used to computer dust fluxes. For this purpose, we selected the YSU Non-local-K scheme (Hong and Pan 1996, Hong et al. 2005).

For an unstable boundary layer, it non-locally adjusts the vertical shear and vertical gradient of potential temperature (as produced by WRF at the resolved grid) back to neutrality. In the process, the scheme estimates the required turbulent momentum and heat fluxes for such restorations. Friction velocity, which is related to the square root of the covariance of turbulent velocities, is produced by the model as a byproduct.

2.6 Land Surface Model and Urban Canopy Model

The WRF model has a land surface module embedded in it to process the interaction among the atmosphere, land surface, and sub-surface soil layers. This module, called *Noah land surface model* (Chen, et al., 2011), calculates the surface fluxes of heat, moisture, momentum, and simulates the evolution of soil moisture according to a diffusion process and the source and sink due to precipitation and evaporation. Figure 7 schematically shows the relevant processes in the model. In our numerical experiment with urbanization, the changes in land cover will lead to changes in all surface fluxes as computed by the land-surface model, the meteorological fields in the atmosphere as computed by the WRF dynamical core, and the detailed interactions between the two.

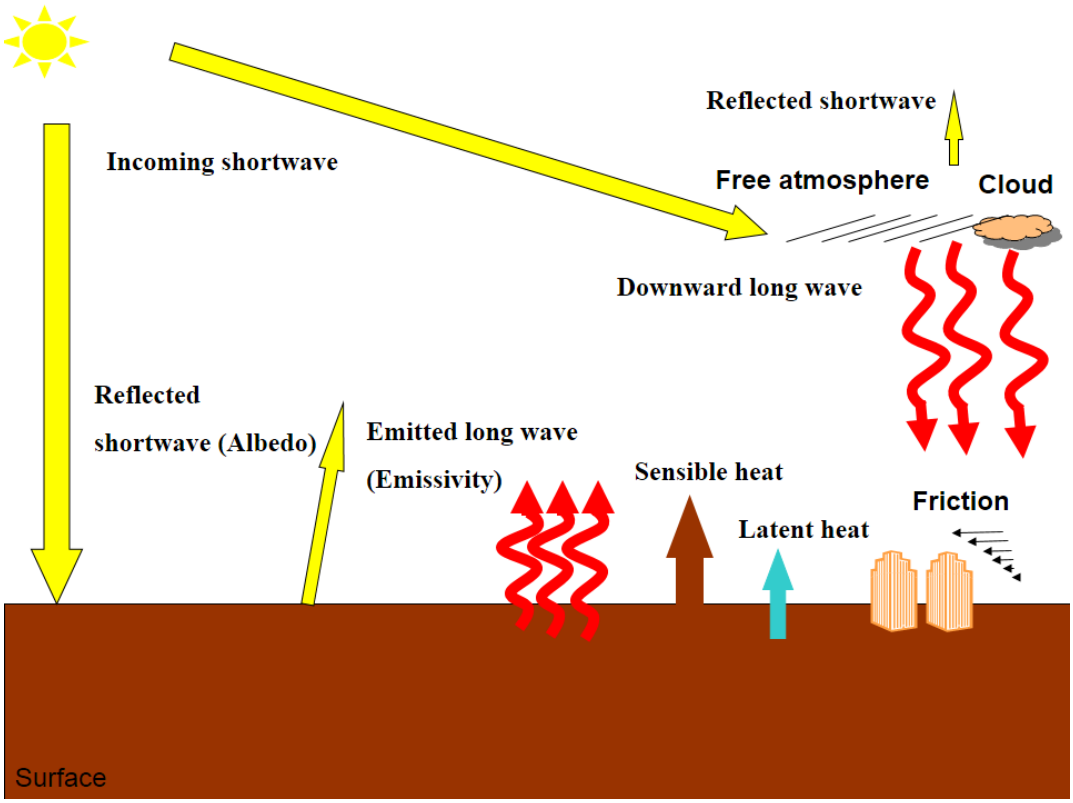

Figure 7: Schematic interaction processes of land surface and radiation models (Kamal, 2015a)

Among different land surface types, the urban type is special because the presence of this land type implies not only a change in the material of the surface but also an effective increase in the surface area and surface roughness due to presence of buildings. These additional changes, and the resulted changes in the characteristics of the flows in the boundary layer over urban areas, are represented by an add-on module (on top of the existing land-surface model) called an Urban Canopy Model (UCM). In our simulations, we also activate the UCM in WRF. Specifically, we choose the single-layer UCM developed by Kusaka et al. (2001) and Kusaka and Kimura (2004) for this purpose.

The UCM allows users to set urban parameters such as building heights, roughness length above urban canyons (for momentum equation), urban fraction, heat capacity of the material for the surfaces (e.g. roofs and walls). For our numerical experiments, since the details of those parameters are not easily obtained, we used a relatively generic setting similar to that used by Kamal et al. (2015b, 2017) for desert cities. For example, building height, urban fraction, and heat capacity of roofs are chosen to be 7.5 m, 0.9, and $1.0*10^6$ J/m^{3}K, respectively.

2.7 Calculation of Dust Fluxes

The preceding sections describe the processes that use WRF simulations and additional computations by the land-surface and turbulence models embedded in WRF to generate near-surface turbulence parameters (most relevantly, the friction velocity u^*) and soil moisture. Our final step was to use these outputs to compute the dust fluxes. Qualitatively, a high surface wind (or strong turbulence) and low soil moisture (i.e., dry land) should lead to more dust generation. In addition, the amount of dust production depends on the type of surface. To put these factors together for the computation of dust fluxes, we sought empirical relations established by laboratory experiments. Our approach here was influenced by previous studies (e.g., Choi and Fernando 2008) which focused on short-term predictions of air pollution. Also see related surveys in Choi and Fernando (2008) and Laurent et al. (2009).

Based on experimental work, Westphal et al. (1987) proposed that vertical mass flux (D_f) for dust particles with radius less than 10μm can be computed as a function of turbulent friction velocity u^* as

$$D_f = 10^{-14}\, u^{*4} \quad \text{, when } u^* \geq u^*_t \text{ , for predominantly silt and clay soils ,} \qquad (11)$$

$$D_f = 2 \times 10^{-13}\, u^{*3} \quad \text{, when } u^* \geq u^*_t \text{ , for predominantly sandy soils ,} \qquad (12)$$

where u^*_t is a threshold value (to be explained shortly). The above relations were determined from laboratory (wind tunnel) experiments in which the land surface has a uniform physical characteristic.

In applications, particularly in WRF simulations which adopt the USGS 24-category land cover classification (see Chapter 3), a particular land-use category consists of a hybrid of several physical land-cover types. As such, Eqs. (11) and (12) need to be modified with a "coverage factor" R for each land-use category, namely

$$D_f = (1 - R)\, 0.13 \times 10^{-14}\, u^{*4} \quad \text{, when } u^* \geq u^*_t \text{ , for predominantly silt and clay soils,} \qquad (13)$$

$$D_f = (1 - R)\, 0.13 \times 10^{-13}\, u^{*3} \quad \text{, when } u^* \geq u^*_t \text{ , for predominantly sandy soils ,} \qquad (14)$$

The threshold value of friction velocity in Eqs. (13)-(14) should depend on surface roughness and soil moisture. (With a wet surface, a greater surface wind is required to blow dust off the surface.) Relevant semi-empirical relations were developed by Marticorena et al. (1997), Fecan et al. (1999), and Gong et al. (2003). First, a baseline value of u^*_t is given as

$$u^*_{t1} = 0.30\, e^{7.22\, Z_0} \tag{15}$$

where Z_0 is the surface roughness length which depends on the land surface type. Then, the threshold value, u^*_t, in Eqs. (13)-(14) is computed from

$$u^*_t = u^*_{t1} \quad , \quad when \quad w < w' \tag{16}$$

$$u^*_t = u^*_{t1}\,[1+1.21(w- w')^{0.68}]^{0.5} \quad , \quad when \quad w \geq w' \quad , \tag{17}$$

where w is volumetric soil moisture, and $w' = 0.0014(\%clay)^2+0.17(\%clay)$. In our simulations, for the formula of w' we assumed a 50% clay content. (This can be refined in future work). With this setting, $w' = 0.08538$ m^3/m^3. The formulas used here do not provide the detailed size distribution of the dust. The detail of size distribution requires further information from either laboratory experiments or field measurements. Thus, we restrict our calculations to the bulk (total) dust fluxes.

The values of the static parameters, R and Z_0, depend only on the land-use categories. They are listed in Table (1) (after Park and In 2003, and Choi and Fernando 2008) for the 24 standard USGS LULC categories.

Table 1: US Geological Survey land-use land-cover (LULC) categories and the corresponding surface roughness length (Z0) and the dust emission reduction factor (R). (Source: USGS. The table is adapted from Choi and Fernando 2008.)

No.	Description	Z_0(cm)	Reduction factor (R)
1	Urban and built-up land	100	1.0
2	Dry cropland and pasture	2	0.4
3	Irrigated cropland and pasture	2	0.6
4	Mixed dry/irrigated cropland and pasture	2	0.5
5	Cropland/grassland	2	0.5
6	Cropland/woodland	2	0.7
7	Grassland	2	0.6
8	Shrubland	3	0.7
9	Mixed shrub/grassland	3	0.75
10	Savanna	2	0.8
11	Deciduous broadleaf forest	5	0.9
12	Deciduous needleleaf forest	5	0.9
13	Evergreen broadleaf forest	5	0.9
14	Evergreen needleleaf forest	5	0.9
15	Mixed forest	5	0.9
16	Water	0.1	1.0
17	Herbaceous wet land	0.2	1.0
18	Wooded wet land	0.3	1.0
19	Barren or sparsely vegetated land	1	0.1
20	Herbaceous tundra	0.3	1.0
21	Wooded tundra	0.3	1.0
22	Mixed tundra	0.2	1.0
23	Bare ground tundra	0.1	1.0
24	Snow or ice	0.1	1.0

CHAPTER 3

INCORPORATION OF LAND-USE CHANGES IN MODELS

3.1 Overview and Focus Location

For the main simulations in this study, we intend to select a desert city that has experienced very rapid urbanization in the last few decades. Moreover, an ideal choice for our numerical study was a city that (i) is surrounded by relatively simple pre-urbanization land cover, and (ii) is located over smooth topography so as to minimize the effect of complex terrains. For the convenience of numerical experiments, it is also ideal that there are no other major cities in the vicinity of the target city. Otherwise, the analysis will be complicated by the presence of more than one city in the computational domain. Lastly, reliable data should be available to determine the extent of urban land cover in the pre- and post-urbanization eras.

Based on those considerations, we select Erbil in northern Iraq (approximately 36.2°N and 44.09°E) as our target city, and its surrounding region as the model domain. The WRF model domain, with 3 layers of nesting and with the innermost domain centered at Erbil, is shown Fig. 8. For our first set of simulations, the horizontal resolutions for the three domains are 25 km, 5 km, and 1 km. The linear dimensions of the three domains are approximately 1500 km, 300 km, and 60 km. Twenty-eight vertical levels are used in all three domains, with the top of the model set to p = 50 mb (above the tropopause, at approximately 20 km height). The vertical η-levels are non-uniform in height, but the lowest level is at approximately 27 m height from the surface.

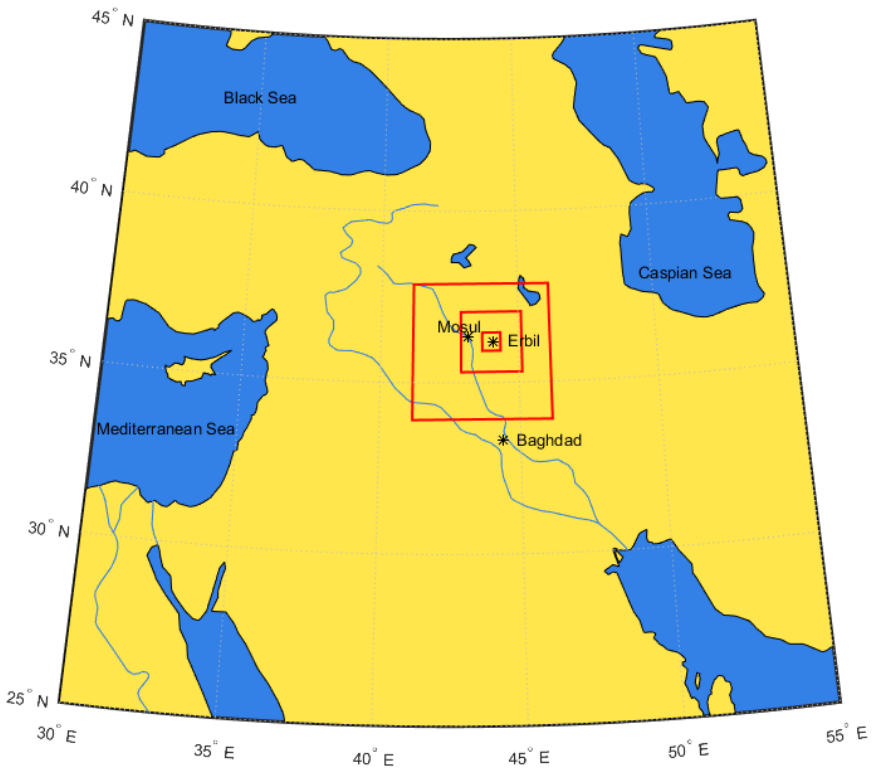

Figure 8: The outermost, intermediate, and innermost domains used for the nested simulations for Erbil city, which is located at the center of the innermost domain.

Figure 9 shows the images from Landsat satellite observation of the city of Erbil and its vicinity for 1987 (left) and 2011(right). The dark gray pixels correspond to urban land cover. This shows that rapid urban expansion occurred between 1987 and 2011. We will choose these two years to define the pre- and post-urbanization eras.

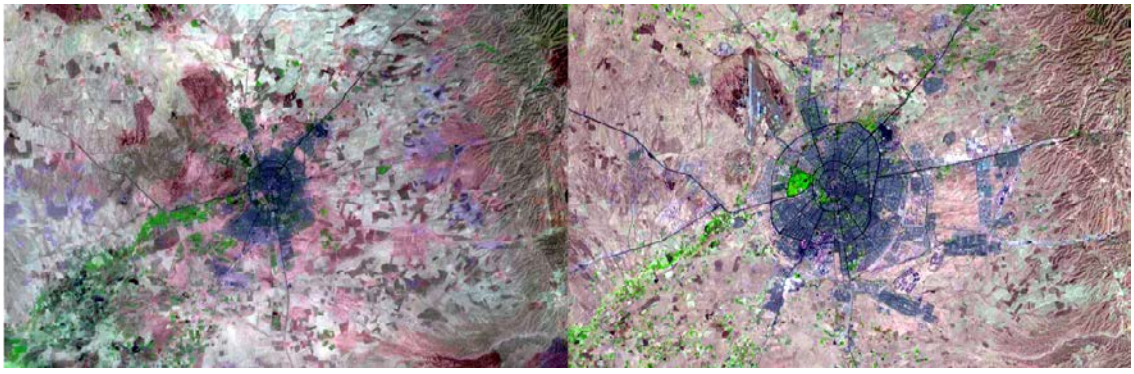

Figure 9: Landsat satellite images for the city of Erbil and its vicinity in 1987 (left) and 2011(right). For reference, in each panel the (longitude, latitude) of the upper left corner is (43°29'E, 36°59'N), and lower right corner (45°03'E, 35°06'N). (Source:USGS.)

Urban expansion over Erbil was generally associated with a conversion from arid types of land to simple urban-type of land, the latter dominated by concrete and low-rise buildings (Rasul 2016). This justifies our choice of using a relatively simple urban land type (see Sec. 3.2) to represent all "urban" grid cells in WRF. Moreover, the relatively low building heights justifies our choice of using the setting of 7.5 m as average building height in the Urban Canopy Model.

3.2 Land-Use Land-Cover (LULC) Maps

Next, we used detailed city maps of Erbil from the pre- and post-urbanization eras to set the surface boundary condition in WRF. First, note that the WRF package comes with a generic land-use map (using multiple sources without a clear association to a particular year) which contains almost no urban land over Erbil, as shown in Fig. 12(a). As such, our strategy was to first use the satellite maps from Landsat observations (which was also cross-validated with published geographic maps for Erbil) to determine the WRF grid boxes that are covered by urban land in 1987 and 2011 (Faqe Ibrahim, 2015). Those were then used to replace the default generic land-use map in Fig. 10(a), only over the region that cover the modern metropolitan area of Erbil. Elsewhere, the default land-use map in WRF is retained. Note that this design is consistent with our idea of running twin experiments with the only difference between the two runs coming from the surface boundary condition related to urbanization.

Figures 10(b) and (c) show the land-use maps (over the innermost domain of WRF) for the 1987 and 2011 eras constructed from this procedure (Codes for facilitating the changes in land-surface, Appendix B, I). They are used in the actual WRF

simulations. In those maps, the dark blue color corresponds to urban type of land. Thus, it is clear that the urban area grew rapidly from 1987 (~ 94 km^2) to 2011 (~ 284 km^2). A few bright yellow grid boxes that emerge in western Erbil in 2011 are associated with a new park that was constructed between 1987 and 2011. Over there, the land surface type is mixed grassland and shrubland instead of urban. The preceding setups completed our construction of the surface boundary conditions for the planned WRF simulations.

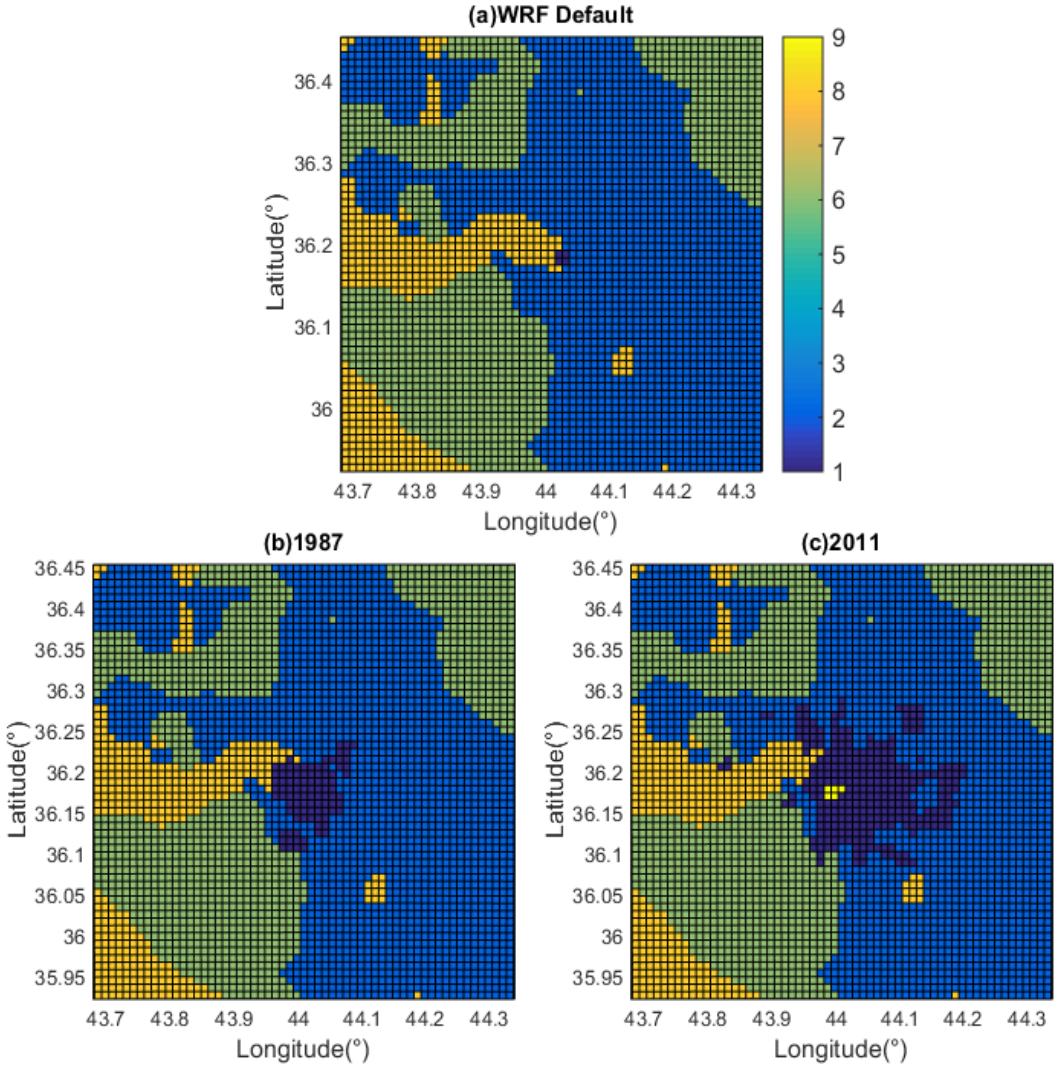

Figure 10: The land-use map over the innermost domain of WRF. The size of each grid cell is 1 km x 1 km. The City of Erbil is located at the center of the domain. (a) The default land-use map in WRF Geogrid database. It is generic and does not have an accurate representation of Erbil. (b) Land-use map for 1987 constructed from satellite images and local surveys. (c) Similar to (b) but for 2011. The maps in (b) and (c) are used to override the default map in in the WRF. Colors represent the LULC categories: 1-Navy blue- Urban; 2-Blue- Dry cropland; 6-Green-Cropland/woodland; 8-Tan: Shrubland; 9-Bright yellow-Mixed shrubland/ grassland. A new park established in western Erbil in 2011 is represented by Category 9.

CHAPTER 4

CHANGES IN METEOROLOGICAL FIELDS DUE TO URBANIZATION

4.1 Simulations for Erbil in Pre- and Post-Urbanization Eras

We have completed a series of WRF simulations for a multiply nested domain centered at Erbil. The detailed changes in land cover from the pre- to post-urbanization eras were given in Chapter 3. To extract the effect of urbanization, we performed "twin experiments" with identical external forcing (by solar radiation) and boundary conditions except the land-use maps in the surface boundary condition. For this purpose, the lateral boundary conditions were constructed from the large-scale observations for a generic year, chosen as 2000. Hereafter, the two sets of runs for "1987" and "2011" refer to the simulations with the same generic lateral boundary conditions but different land-use maps deduced from 1987 and 2011, respectively.

Since dust generation depends on the meteorological conditions which have a strong seasonal dependence, we conducted two runs for each of the 1987 and 2011 cases. The "summer" simulation lasts for 30 days in the month of July (from July 1-30), and "winter" simulation 30 days in the month of January (from January 1-30) Table 2. The output of major meteorological variables (e.g., 3-D wind, temperature, and precipitation), friction velocity produced by the turbulence model, and soil moisture produced by the land-surface model, are all saved at a 3-hour frequency. (This is much longer than the 3-minute time step but is adequate for resolving the diurnal variation of meteorological fields.) Each run has the total of 240 3-hourly outputs which we used for further analyses.

Table 2: A summary of the main WRF simulations

Run #	Land Use Land Cover for Surface B.C.	Lateral B.C. at the outer most domain	Output Frequency (hrs.)	Duration (Days)
First	Erbil 1987	6-hourly global FNL January 2000	3	30
Second	Erbil 2011	6-hourly global FNL January 2000	3	30
Third	Erbil 1987	6-hourly global FNL July 2000	3	30
Fourth	Erbil 2011	6-hourly global FNL July 2000	3	30

4.2 Analysis of Meteorological Fields

4.2.1 Analysis of 2m Temperature Fields

Previous studies on the effect of desert urbanization have focused on surface air temperature. The well-known “urban heat island” effect was shown to generally hold for desert cities in nighttime (e.g., Kamal et al. 2015b, 2017, Rasul et al. 2015), although this is accompanied by a weak daytime cooling (e.g., Kamal et al. 2015b, 2017). Since the nighttime warming is considered a robust signal, we first use it as a basis to check our simulations for Erbil.

Following previous studies, we compare the 2 m air temperature from the 1987 and 2011 simulations. Figure 11 shows the time series of 2 m air temperature averaged over a box that approximately covers the extent of the urban area of modern Erbil. Shown is the 3-hourly data for the whole month of (a) January and (b) July, with the 1987 and 2011 cases colored in black and red. In both figures, even on a day-to-day basis one can observe a systematic increase in nighttime temperature from 1987 to 2011 while the difference is muted in daytime. (However, the signal of daytime cooling becomes clearer after an average over the month, as shown below). This is consistent with the findings from previous studies for different desert cities.

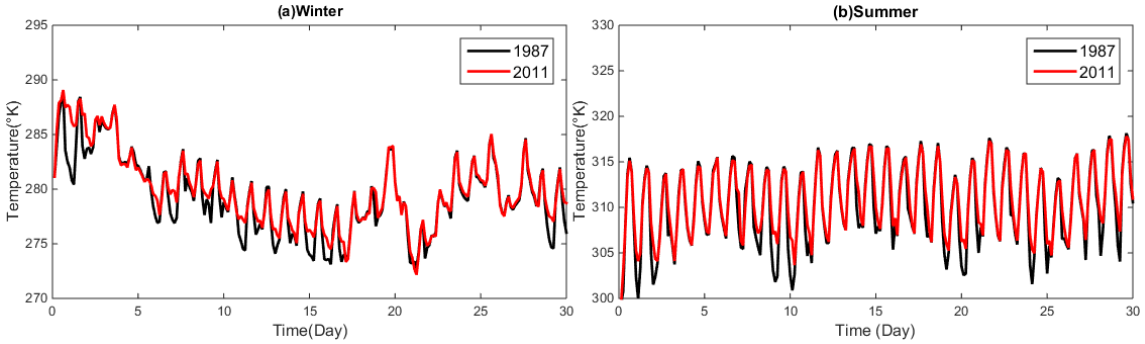

Figure 11: Time series of air temperature at 2 m height averaged over Erbil. Shown are 3-hourly model output over a 1-month period. The lateral boundary conditions are from 2000, and red and black are for the simulations using 2011 and 1987 land-use map, respectively. (a) Winter (January); (b) Summer (July).

Figure 12 shows the (Innermost model domain centered at Erbil) maps of the difference, defined as "2011 case" minus "1987 case", in nighttime 2 m temperature averaged over the entire months of January and July, respectively (Left to right). Warming over the newly established urban area is readily identified in the upper two images (a) and (b). We used the average of midnight and 3 AM local time to define "nighttime" while the maximum of nighttime warming usually occurs in early morning.

Thus, the 1.5 °*C* and 2°*C* difference shown in the two figures would increase slightly if a later time is chosen). The magnitude of warming is also consistent with the estimates from other studies for desert cities.

The lower two maps (c) and (d) of Fig. 12 represent daytime 2 m air temperature. Consistent with previous studies, a weak but robust signal of daytime cooling due to urbanization emerge over the newly established urban areas. Note that in this study, as well as in previous numerical studies cited before, the daytime cooling arises mainly from an increased effective surface area over the city (due to presence of buildings; this is equivalent to the "shadow effect", e.g., Kamal et al. 2015b). Also, note that anthropogenic effects such as increased irrigation within the city, or anthropogenic production of heat, are not explicitly included in our study.

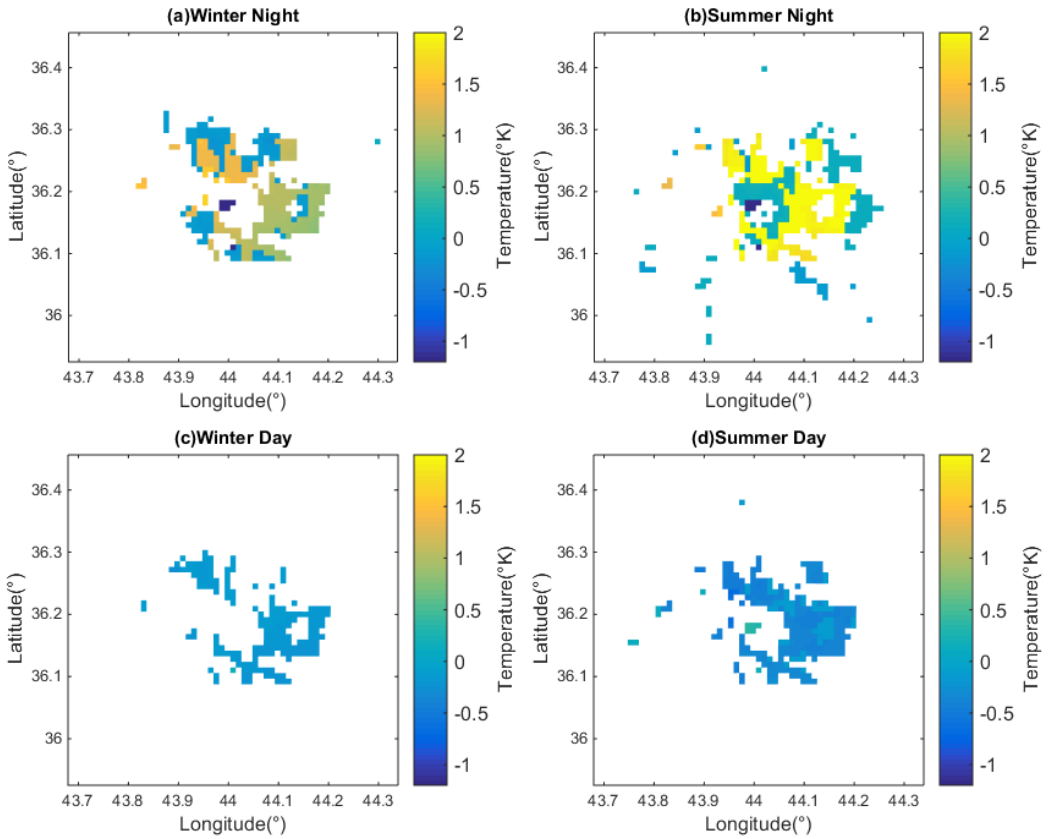

Figure 12: The difference in 2m temperature, defined as "2011 case minus 1987 case", that represents the effect of urbanization. Shown are the monthly averages over Erbil and its vicinity. Panel (a) and (b) are nighttime temperature for winter and summer, showing a warming effect of approximately 1.5°K and 2°K, respectively. Panel (c) and (d) are daytime temperature for winter and summer, showing a slight cooling effect of approximately 0.25°K and 0.45°K, respectively.

In both cases the intensity of cooling and warming patterns were higher in summer than winter, which is also generally consistent with previous studies. Lastly, note that in all images of Figure 12 there is a spot in the western side of the city over which the temperature difference is the opposite of that over the newly established urban areas. That spot is the site of a large park covered by grasses and shrubs. As the park was established between 1987 and 2011, over the site there was no urbanization, but the former desert land was converted to grassland. This leads to the different characteristics in the temperature signal.

4.2.2 Analysis of 10 m Wind Fields

A relatively small number of studies have examined the effect of urbanization on the near-surface winds. Kamal et al (2015a) found a reduction of the strength of diurnal circulation over Las Vegas (a desert city) due to urbanization. Rajagopalan et al. (2014) found a similar reduction of low-level wind speed over a tropical (non-desert) city. The changes in wind field could, in turn, affect the temperature by altering the "ventilation effect" (e.g., Kamal et al. 2015a). To compare with existing works, Fig. 13 shows the 3-hourly time series of the 10m wind speed averaged around Erbil, for winter and summer. Comparing the 1987 (black) and 2011 (red) cases in Fig. 13, we see a slight reduction of wind speed in 2011, consistent with previous studies. The reduction can be visualized more clearly in the maps of the wind speed, shown in Fig. 14. Over most of the areas where urbanization occurred between 1987 and 2011, we find a decrease in surface wind speed.

The analysis in this chapter affirms that our results of the basic meteorological fields from the numerical simulations of a desert city are broadly consistent with previous studies.

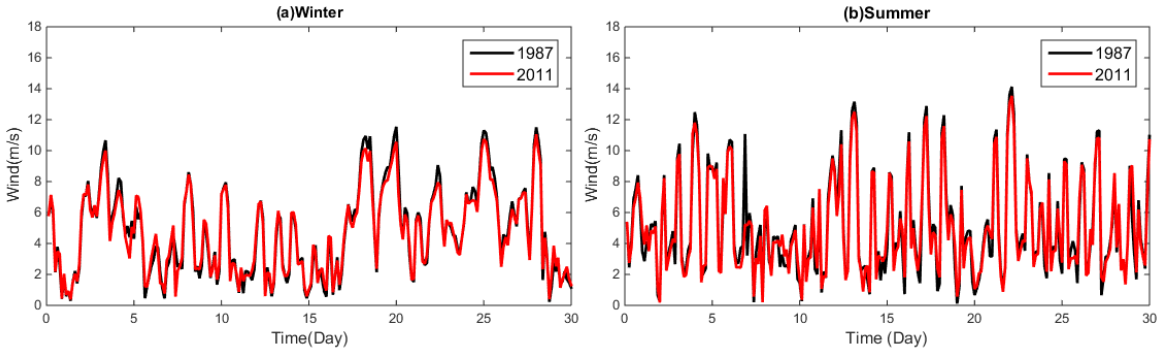

Figure 13: Time series of surface wind speed at 10 m height averaged around Erbil. Shown are 3-hourly values over a month. The black and red curves are for the 1987 and 2011 cases, respectively. (a) Winter (January); (b) Summer.

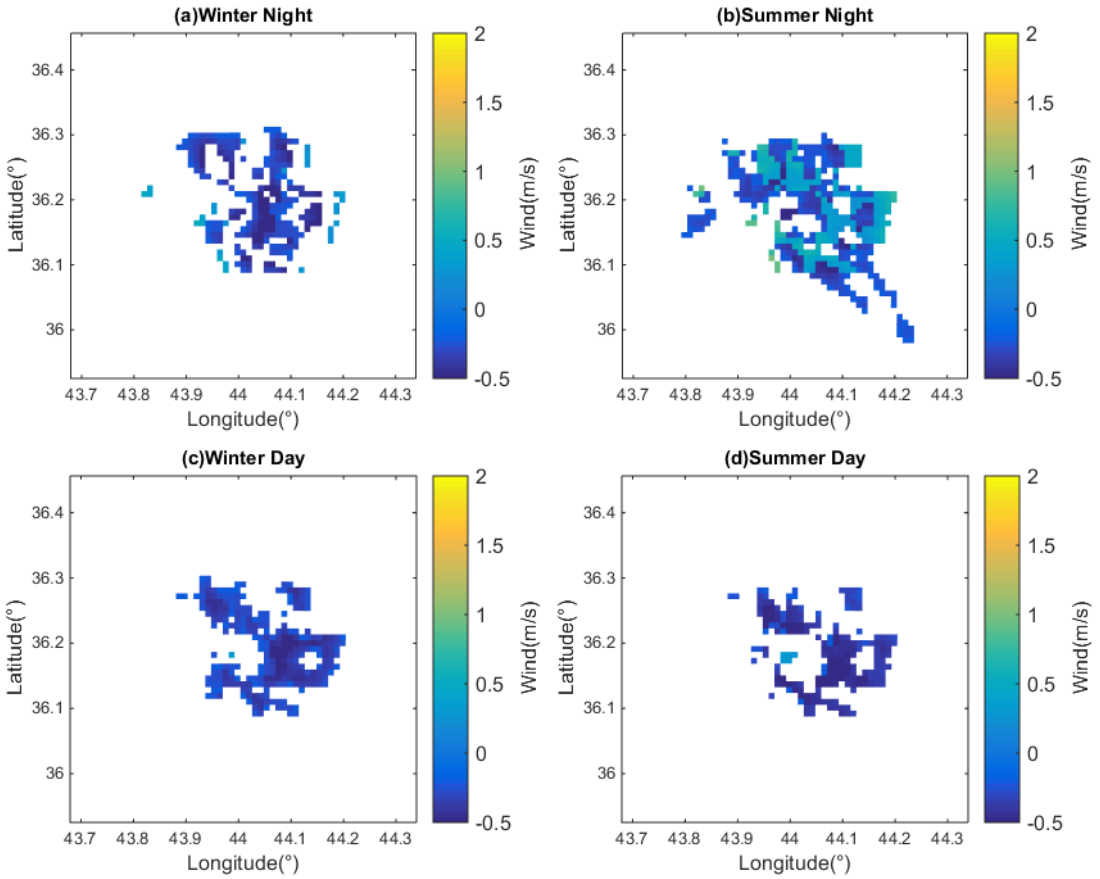

Figure 14: The differences in 10 m wind speed, arranged in the same fashion as Fig. 12. All 4 panels show a slight reduction in the wind speed on the order of 0.25-0.5 m/s.

CHAPTER 5

VALIDATION WITH SELECTED OBSERVATIONS

5.1 Choice of Observational Data

In desert regions, in-situ observations at meso-scale resolution are relatively scarce. Over the areas and time period with little availability of real observations, the output from our numerical simulations could be used as "virtual observations" to fill the data gap. In fact, this is part of the motivation of this study. Nevertheless, we will attempt to validate the meteorological fields from the numerical simulations for Erbil with selected observation data when they are available.

A search of archived data of in-situ observations of meteorological variables over Erbil, including communications with local city officials, did not return concrete information. Although limited data for temperature and wind are available from local archive, they lack a precise description of where and by what methods the variables were measured. This makes it difficult to meaningfully compare the data with numerical simulations. Instead, we choose to perform the validation using a reanalysis dataset, which is produced by assimilating raw meteorological observations (from in-situ surface station measurements, upper-air sounding, and remote sensing by satellite). At major meteorological centers, this is done by feeding the raw observations with irregular spatial and temporal resolutions into a gridded global model to initialize a very short run. In the process, all dynamic and thermodynamic variables undergo quality check and mutual adjustments. The outcome is a global gridded dataset for the meteorological variables. The reanalysis data serves as a useful proxy of the raw observation. Specifically, we

choose to use the CFSR reanalysis (Saha et al. 2010), produced by the National Centers for Environmental Prediction (NCEP) of the United States.

Our key simulations for Erbil are constrained by the lateral boundary conditions from 2000. The "1987" and "2011" cases are the runs with the default land-use map in the model replaced by that for 1987 and 2011. As illustrated in Fig. 15, the land-use map over Erbil in 2000 is actually very close to that in 1987, as rapid urbanization did not start until after 2000. With this, we take a short cut by comparing the numerical simulation of the "1987" case with the reanalysis from 2000. The comparisons are carried out for winter (January) and summer (July). We choose the version of CFSR reanalysis with 0.5° *x* 0.5° resolution. Four grid points from the dataset fall within the innermost domain of WRF. The comparisons will use the data on two of them that are located to the south of Erbil, marked by *L1* and *L2* in Fig. 18. Superimposed in the figure is the pre-urbanization land-use map over Erbil. The latitude and longitude of the CFSR grid points are indicated in Fig. 16. The elevation of *L1* and *L2* are 287 m and 439 m, respectively. Since temperature depends on elevation, the comparisons are made between the CFSR data and the model output averaged over only a few grid points of WRF in the close vicinity of the CFSR grid point.

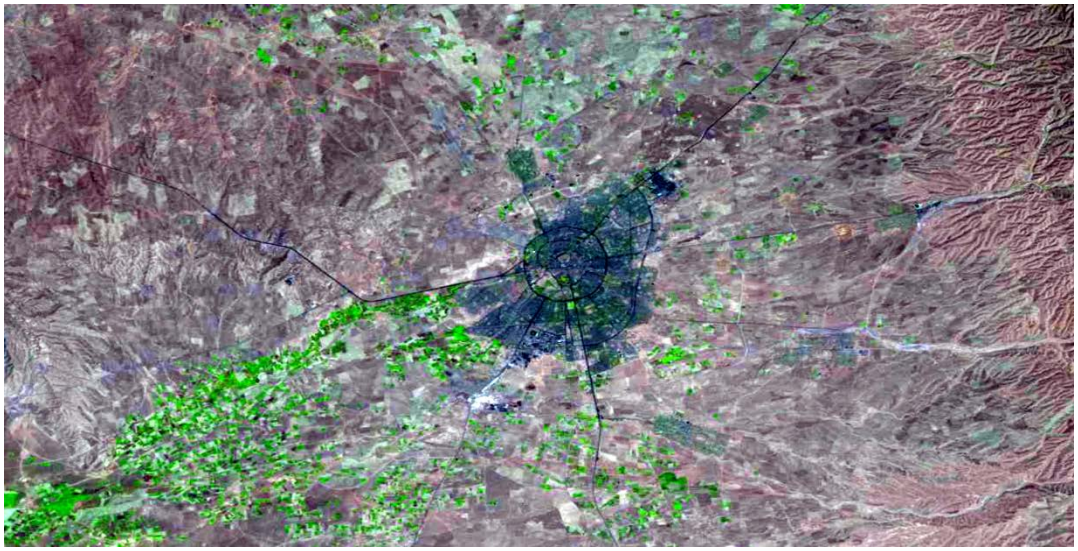

Figure 15: Landsat satellite image over the region in the vicinity of city of Erbil from 2000. The domain shown is similar to that in Fig. 9. (Source: USGS.)

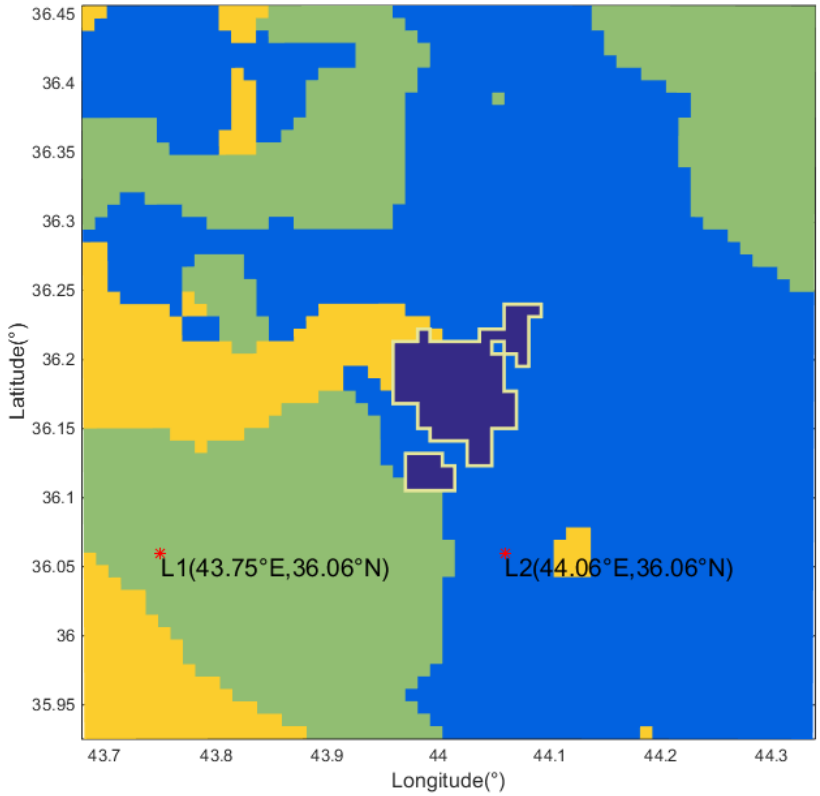

Figure 16: The locations of the two grid points of CFSR, marked by L1 and L2, where the comparisons between WRF simulations and reanalysis are performed.

5.2 Comparisons for Temperature, Humidity, and Precipitation

In all the ensuing comparisons between the CFSR reanalysis and WRF simulations, we keep in mind that the WRF model has much higher spatial and temporal resolutions relative to those in the CFSR global model used for reanalysis. For example, for upper-air variables the reanalysis is constrained mainly by the global twice-daily (or 6-hourly in limited number of locations) sounding observations. The detail (e.g., hour-to-hour variation) of the diurnal cycle in the reanalysis is somewhat artificial as it is generated by the CFSR model. As such, we compare only the daily-averaged data. Figure 17 (a) and (b) show the time series of daily-averaged 2 m air temperature from WRF simulations and CFSR reanalysis, at the *L1* and *L2* sites for winter (January). The WRF simulations do capture the overall temporal variation in temperature, with an approximately 1.5°*C* cold bias at site *L2*, possibly due to the difference in the topography of WRF and CFSR models. Since the WRF simulations are constrained by the FNL data only at the boundary of the outermost domain, this affirms the success of dynamical downscaling through successive layers of nesting in the WRF simulations.

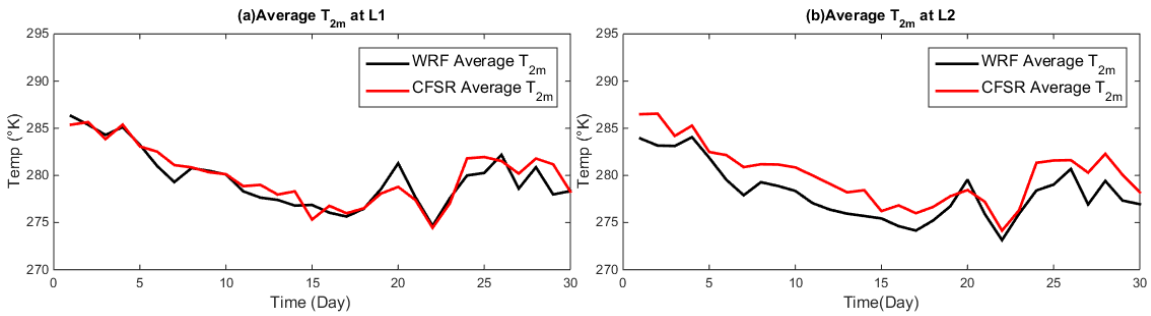

Figure 17: Time series of daily-averaged 2m Temperature over a month. Red and black curves in each panel are WRF and CFSR, respectively. (a) L1; (b) L2.

Comparisons for the daily averaged relative humidity, arranged in a similar fashion as Figs. 17 (a) and (b), are shown in Fig. 18 (a) and (b). Like the daily averaged temperature, the agreement between WRF and CFSR reanalysis is reasonable. The difference between the two appears to be random instead of systematic, although at site *L2* WRF appears to be slightly drier overall.

Comparisons for the daily precipitation are shown in Figs. 19 (a) and (b). The timing of strong precipitation events as produced by WRF is consistent with that from CFSR. The WRF simulation underestimates the amount of rainfall for the weak events, but overestimates it for the strong events. Previous studies have shown a similar bias in WRF of excessive rainfall for extreme events when the horizontal grid size is refined to below ~ 6 km (e.g., Sharma and Huang 2012).

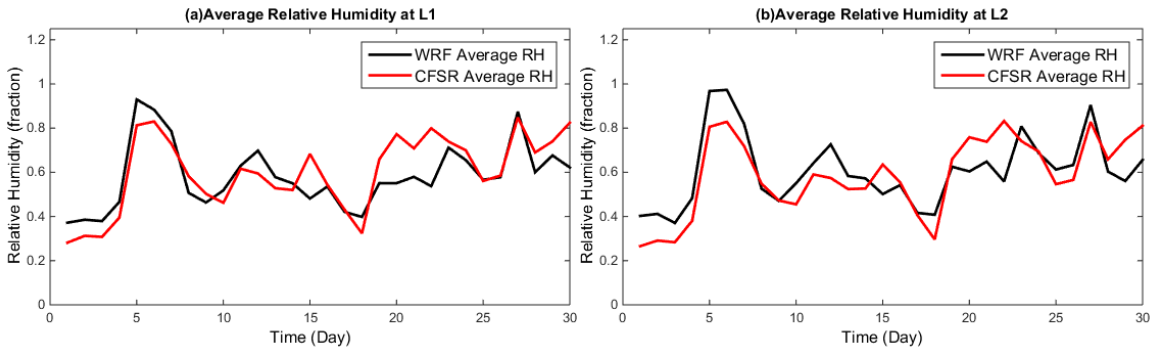

Figure 18: Time series of daily-averaged relative humidity over a month. Red and black curves in each panel are WRF and CFSR, respectively. (a) L1; (b) L2.

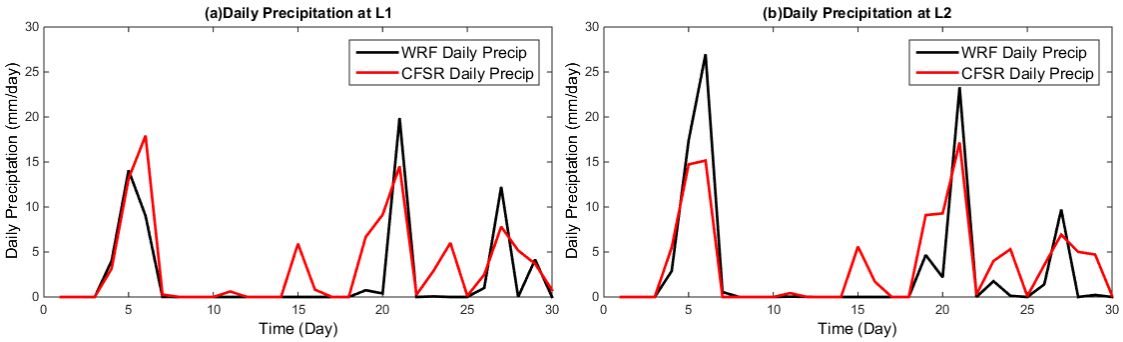

Figure 19: Time series of cumulative daily-precipitation over a month. Red and black curves in each panel are WRF and CFSR, respectively. (a) L1; (b) L2

5.3 Comparison of 10 m Wind Speed

The validation of velocity might be more difficult given that the wind field generally depends more sensitively on model resolution. Recall that WRF has a much higher horizontal resolution (and a more refined topography associated to it) as compared to the CFSR model used to produce reanalysis. Nevertheless, if the velocity in a WRF simulation does not vary greatly in space (see Fig. 27(a) for an example), it could still be meaningful to perform a comparison between WRF and CFSR. Figures 20 (a) and (b) show the daily-averaged 10m wind speed from WRF and CFSR, in a format similar to Figs. 18 (a) and (b). Noticeable differences exist between WRF and CFSR, with the wind speed from WRF systematically stronger. Amplitude aside, the timings of acceleration and deceleration of wind in the time series of WRF and CFSR broadly agree with each other.

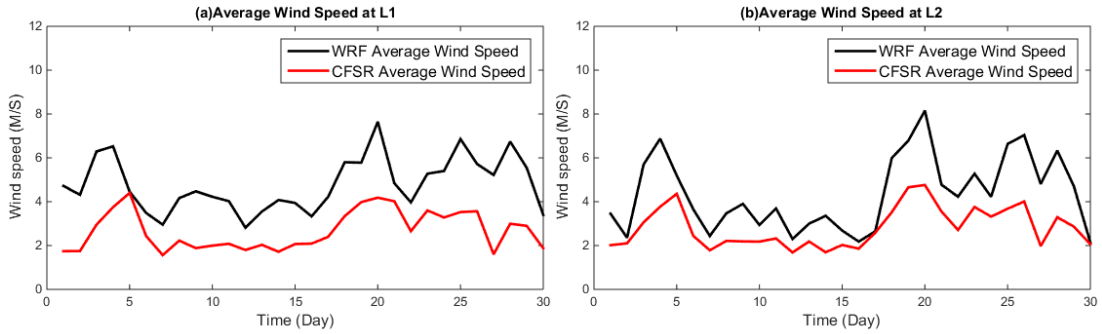

Figure 20: Time series of daily-averaged 10m wind speed over a month. Red and black curves in each panel are WRF and CFSR, respectively. (a) L1; (b) L2.

The results from the inter-comparisons broadly suggest that the local meso-scale features in the thermodynamic fields produced by WRF through dynamical downscaling are reliable. The comparisons reveal a positive bias in precipitation, although the model still produces correct timing for a strong-precipitation event. The bias in precipitation implies a wetter surface in WRF which could lead to an underestimate of dust production. (However, as will be analyzed in Chapter 6, this effect turns out to be minor over the desert-type land in the region of Erbil.) Taking the results in Fig. 20 at face value, the surface wind in WRF is biased to the stronger side, although the overall time evolution of the increase and decrease of surface wind still agrees with reanalysis. The reason for the positive bias in wind speed remains to be investigated. It is useful to note that the topography used in the WRF simulations has a much higher resolution compared to CFSR. It is also possible that some of the strong local winds in WRF are realistic, but they are absent in the CFSR reanalysis as no high-resolution observations of surface wind were available to feed into the data assimilation process for CFSR.

CHAPTER 6

RESULTS FOR DUST GENERATION

6.1 Calculation of Dust Generation

Using the formulas in Section 2.7 from chapter 2, the 3-hourly outputs of friction velocity u^* and soil moisture w from the WRF simulations were used to compute the dust fluxes over the computational domain. Additional information that is required for the calculation, namely, the static data related to land surface types, has been preprocessed as explained in Chapter 3. The calculation produces 3-hourly maps of dust fluxes through the entire month for each simulation. Since both u^* and w vary with time and spatial location, the maps of dust fluxes also show complicated structures.

Figure 21 shows the maps (two on left (a) and (c)) of winter cumulative (integrated over the month of January) dust fluxes over the innermost model domain centered at Erbil, for the city size on 1987 and 2011 cases. The maps (two on right (b) and (d)) of Figure 21 represent summer cumulative (integrated over the month of July) dust fluxes over the innermost model domain centered at Erbil. The areas with minimum dust generation (shown in dark blue) are those over the Erbil city and the strip of grassland to its west. The expansion of the urban area of Erbil creates a greater area with minimum dust generation (bottom two (b) and (d)) of Figure 21. Note that the spatial pattern and magnitude of dust fluxes are significantly different in winter and summer. This shows a strong influence of meteorological conditions on the details of dust generation.

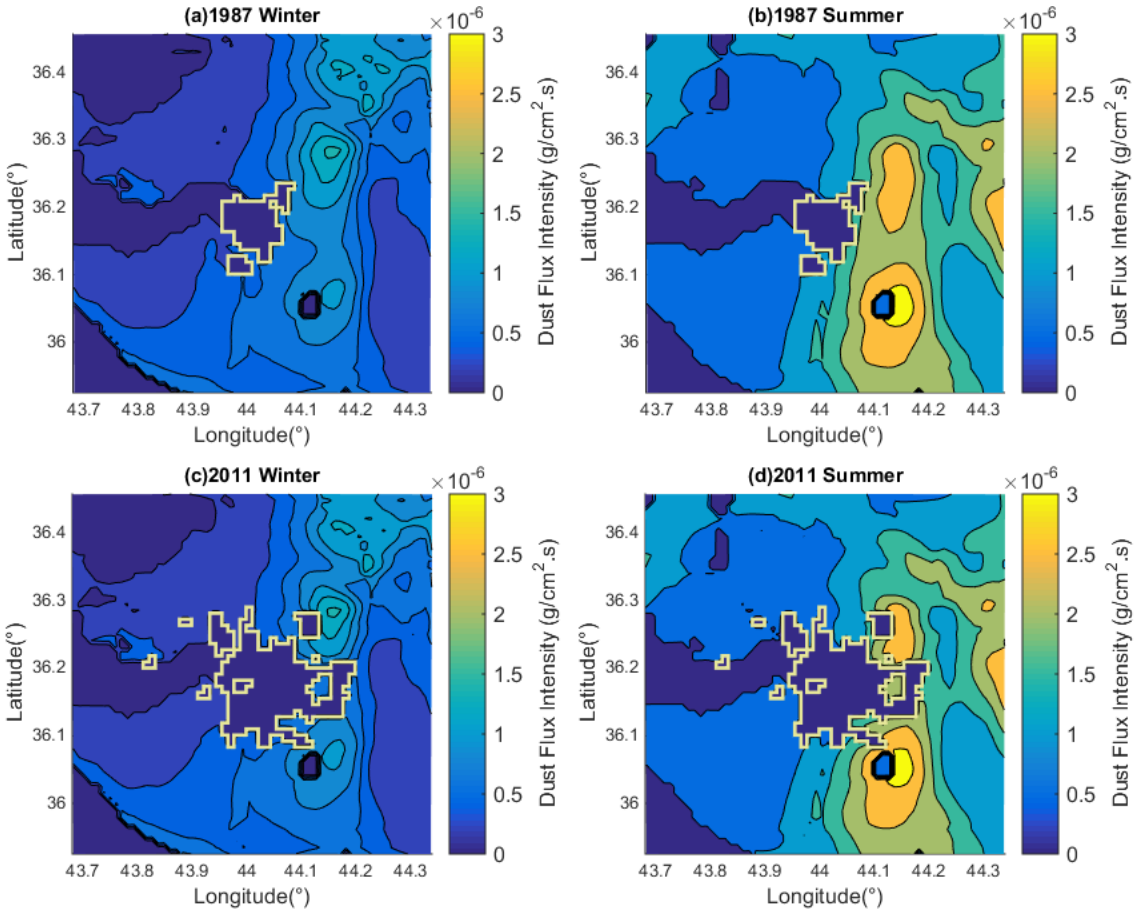

Figure 21: Cumulative dust flux (g/cm^{2}s) over a period of 30 days. Panel (a) and (b) are from the 1987 case, for winter (January) and summer (July), respectively. The border of the urban area in 1987 is marked by light yellow lines. Panel (c) and (d) are similar to (a) and (b) but from the 2011 case. The border of the urban area in 2011 is marked by light yellow lines.

In Fig. 21, the areas with high levels of dust generation are mostly in a zone to the east of Erbil. This is likely due to two reasons. First, wind speed is generally stronger to the east of the city. (An example of the monthly-mean 10m wind vector can be found in Fig. 27 (a).) Also important is that land cover over the region east of the city is dominated by the category of Dry Cropland (*cf.* Fig. 10); The cultivated land has soft soil, implying easy dust generation. In contrast, the land cover to the west of the city is dominated by the categories of Shrubland and Cropland/Woodland, which have less capacity for emission of dust.

Figures 22 show the difference in the cumulative dust fluxes between 1987 and 2011 for winter on left (a) and summer on right (b), respectively. For this particular set of maps, since dust generation is suppressed over the newly established urban areas in 2011, we use the reverse definition of “1987 minus 2011” to define the positive difference. (Thus, a positive value means more dust generation in 1987.)

As expected, the areas with positive differences in dust generation approximately coincide with these that underwent urbanization between 1987 and 2011. Nevertheless, note that even within those areas (where the same type of land conversion occurred) there are nontrivial spatial variations in the magnitude of the differences. Again, this shows the added value of the meteorological model simulations, as the spatial variations of the meteorological fields help create those detailed patterns.

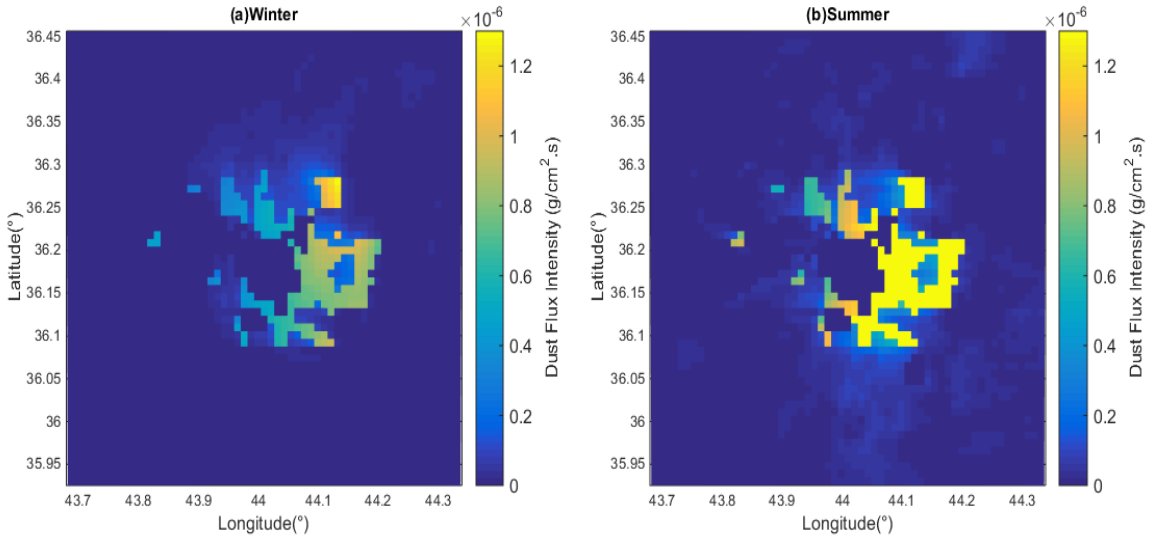

Figure 22: The difference in the monthly cumulative dust flux, defined as "1987 case minus 2011 case". (Positive means more dust is produced in the 1987 case.) (a) winter (January); (b) summer (July).

6.2 Influence of Meteorological Conditions on Dust Generation

The detailed formulas for dust generation given in the previous section quantify the impact of meteorological conditions on the dust fluxes. Qualitatively, over the same land surface type, dust fluxes increase under a stronger surface wind (or turbulence intensity) and a drier surface (i.e., lower soil moisture). The meteorological simulations help us to understand the relative importance of those factors. In order to perform the in-depth analysis, I choose a 1 km *x* 1 km grid box located in the area which was non-urban in 1987 but became urban in 2011. The location (approximately 44.09°E and 36.17°N) was used for this work. This location was a source of dust back in 1987 and was later suppressed by urbanization in 2011 Fig. 23.

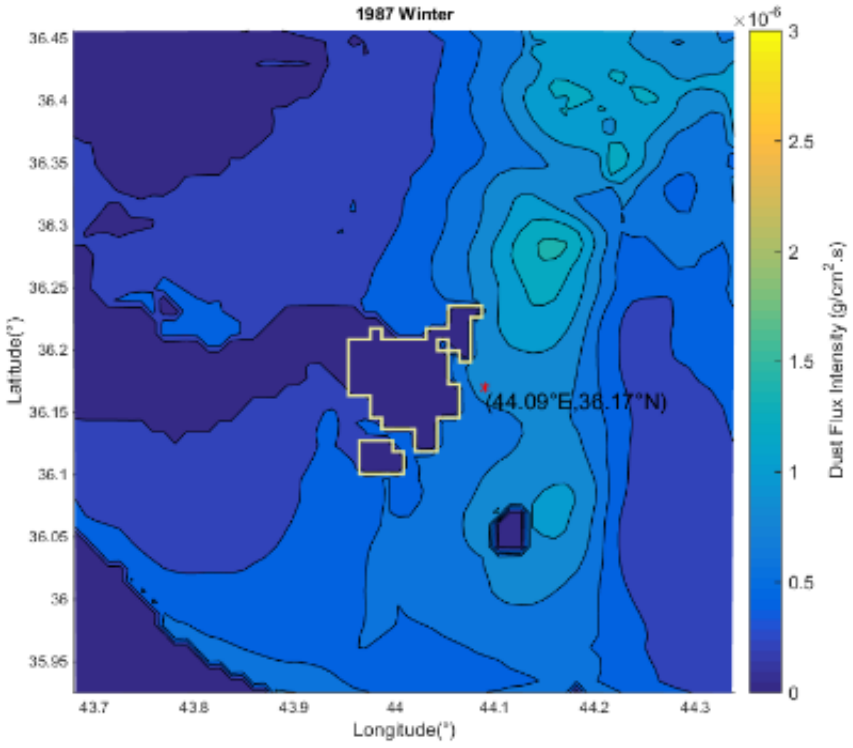

Figure 23: The location (marked by a red star) chosen for a detailed analysis of the meteorological conditions.

Figure 24 (a) shows the 3-hourly time series of the friction velocity (u^*) at a 1 km x 1 km grid box located in the area which was non-urban in 1987, but became urban in 2011. Shown is the winter simulation (over the month of January) for the 1987 case. The value of u^* is shown in black and the threshold value u^*t (to trigger dust production) is shown in red. Figures 24 (b) and (c) are similar to Fig. 24 (a) but for the time series of soil moisture (w) and dust fluxes (D_f). The time series of soil moisture w exhibits dramatic temporal variation with a sharp increase due to a precipitation event, followed by a slow decline. Foe each event, the increase in soil moisture leads to an increase in the threshold value u^*t (as a stronger surface wind is needed to blow dust off a wetter surface), as can be identified in the red curve in Fig. 24 (a). However, for this particular location and season, this effect is minor compared to the variation of u^*. The portion of u^* that rise above the threshold value u^*t leads to dust generation. As such, the temporal evolution of the excessive u^* shares a similar pattern with that of the dust fluxes themselves as shown in Figure 24 (c).

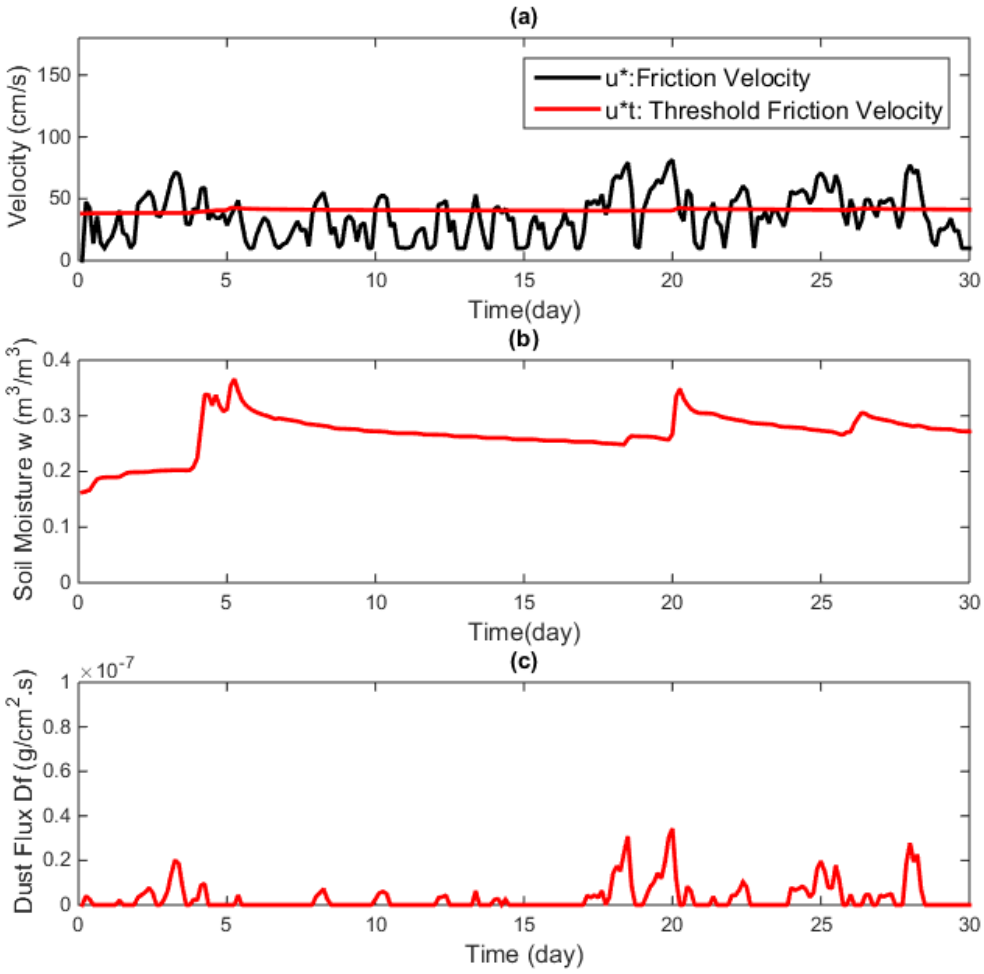

Figure 24: Three-hourly time series over 30 days in January (case 1987) at the chosen site (shown in Fig 22) for (a) Friction velocity, u(black), and the threshold of friction velocity for dust generation, u^*_t (red) (in cm/s). (b) Soil moisture w (m^3/m^3). (**c**) Dust flux ($g/m^2.s$).*

Figure 25 (a, b, and c) plots are the counterparts of above plots but for summer. For this particular location and season, the surface was dry to begin with, and it did not rain over the month. As such, soil moisture continued to decline and the threshold value for *u*t* never changed.

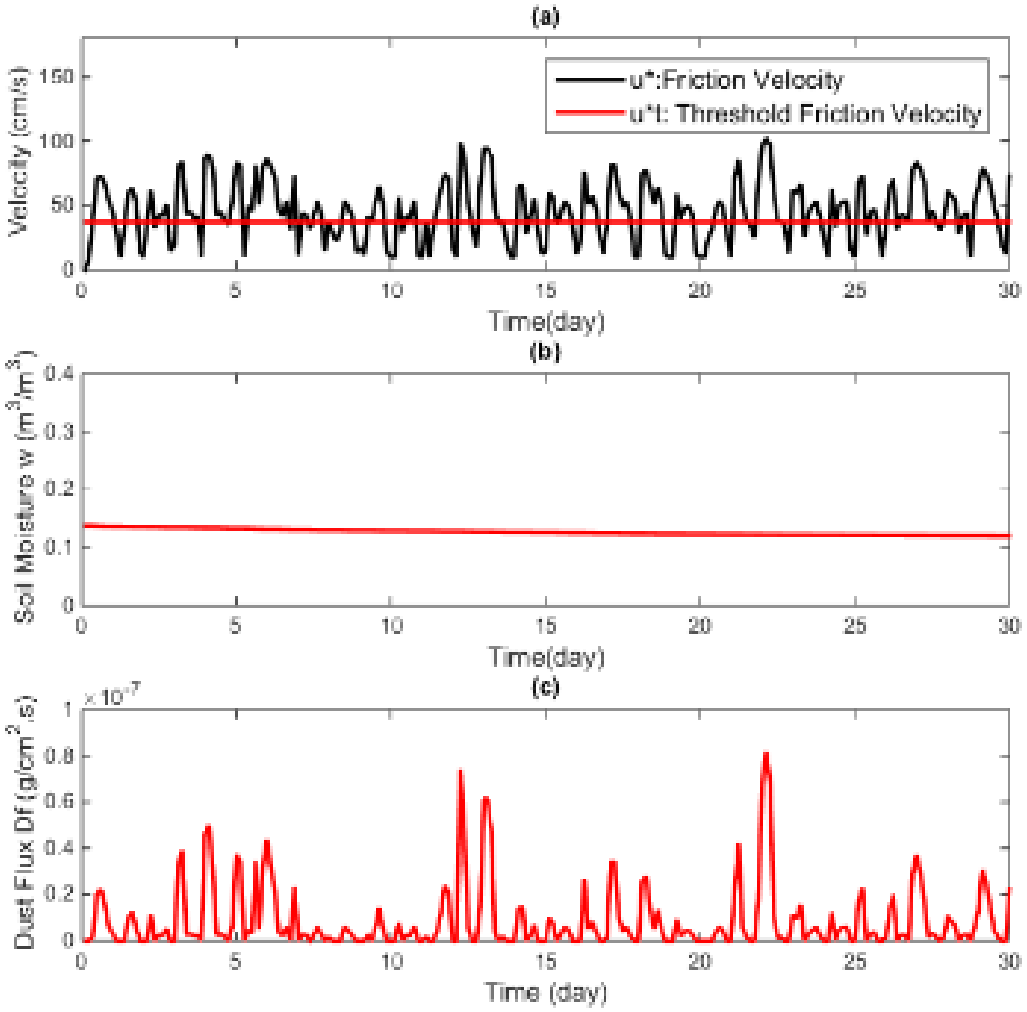

Figure 25: Three-hourly time series over 30 days in July (case 1987) *at the chosen site (shown in Fig 22) for (a) Friction velocity, u*(black), and the threshold of friction velocity for dust generation, u^*_t (red) (in cm/s). (b) Soil moisture w (m^3/m^3). (**c**) Dust flux ($g/m^2.s$)*

6.3 Relation between Climate Change and Dust Generation

The WRF simulations produce detailed 3-D wind and temperature fields, and precipitation, which can be used for further analyses on the correlation between dust fluxes and those meteorological variables. The results shown in Figs. 24 and 25 in previous section demonstrate the potential for such analyses. In future work, one can further expand the existing analysis to systematically correlate dust fluxes with key

meteorological variables across the model domain (note that Figures 24 and 25 in the above section is for only one location). Moreover, the correlations are expected to depend on the baseline meteorological conditions which vary between seasons, and between day and night. For example, the planetary boundary layer is generally more stable at night, which might imply a lesser degree of dominance by turbulence intensity as the factor that controls dust generation. The analyses of the output from our existing simulations will allow us to perform those analyses with seasonal and diurnal contrast, and specifically under the setup of a desert city that undergoes urbanization.

CHAPTER 7

OUTLOOK AND FUTURE WORK

7.1 Transport and Deposition of Dust

In the preceding chapters we successfully quantified the changes in local meteorological conditions and surface dust fluxes induced by urbanization of a desert city. The difference in the dust fluxes could further redistribute in space due to transport by wind. Quantifying the effect of transport will be an important future work. In that context, the surface dust fluxes obtained in this study serve as the boundary condition for the transport problem.

For a medium-sized and isolated desert city such as Erbil, the relevant distance of transport is relatively short. Since the difference in the surface dust flux between the pre- and post-urbanization eras occurs only over a "ring of suburbia" surrounding the old city, the main concern is whether the dust generated there is blown over the city (arrow "*A*" in Fig. 26) or away from the city (arrow "*B*" in Fig. 26). The relevant distance of transport, L, from the source to the target region, is approximately the size of the city. For Erbil, $L \sim$ 10 km. Strong dust generation occurs usually under the condition of strong near-surface turbulence, which usually corresponds to a high wind (for the mean velocity) condition. Taking $U \sim 2$ m/s as the typical velocity scale (*cf.* Chapter 4), the relevant time scale for dust transport is only $T \sim L/U \sim 1$ hour. Nevertheless, for applications, a robust transport model is needed to accurately assess how the excess dust affect life inside the city.

In the following, we outline the main approaches for transport calculations, and related issues with dust deposition in future work:

(1) *Lagrangian transport model*

Since the only relevant "starting points" of transport are those located in the "ring of suburbia", a Lagrangian framework could be relative efficient. With appropriate interpolations for the mean velocity taken from WRF output, forward transport can be computed from the source location as $\boldsymbol{X}_1 = \boldsymbol{X}_{source} + \boldsymbol{V}\,\Delta t$, where $\boldsymbol{X}_{source}$ is the location of dust generation, $\boldsymbol{V}$ is 3-D wind vector interpolated onto $\boldsymbol{X}_{source}$, and the generic symbol of $\boldsymbol{X}$ is understood as the centroid of dust population. Subsequent steps can be performed with $\boldsymbol{X}_{n+1} = \boldsymbol{X}_n + \boldsymbol{V}_n\,\Delta t$, with proper interpolation to obtain *Vn*, and so on. The forward scheme can be improved to semi-implicit to improve stability, but for short-term transport a simpler scheme might be sufficient when a small Δt is chosen. Since *V* comes from WRF output, this can be done by choosing a short output interval of WRF.

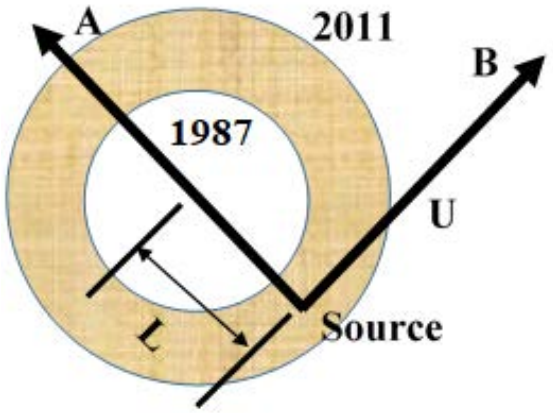

Figure 26: A schematic diagram to illustrate the nature of the problem of short-term transport over Erbil. The concentric circles indicate the extent of urban areas in 1987 and 2011. Differences in dust generation occur over the brown-colored ring. The key question is whether the dust generated over the "ring" is blown over the city (scenario "A") or away from the city (scenario "B"). The relevant length scale for the transport, L, is approximately the linear dimension of the city.

To envision what a Lagrangian calculation might produce given the general climatology over Erbil, Fig. 27 (a) shows the map of monthly-mean 10m wind vectors (for the horizontal wind components) in January from a WRF simulation. Examining the wind fields more closely, we find that the low-level wind around Erbil generally does not have great spatial variations. In winter, the prevailing wind direction is northwestward. Since the main "hot spots" of dust generation are to the east of Erbil, this implies that the most critical spot for dust generation is to the southeast of Erbil, as the dust generated there are more likely to be blown passing the city. Figure 27 (b) shows two examples of the direction and distance of the paths determined by a "one step" calculation of $\boldsymbol{V}\,\Delta t$ from the preceding paragraph, using Δt = 1 hour. At each source site, 4 paths are shown which correspond to 4 starting times 1 hour apart, over a period of intense dust generation in January. The simple diagram in Fig. 27 (b) gives us a broad sense of the relevant time scale (namely, $T \sim 1$ hour is indeed about what we need). At the same time, that the straight "paths" already pass Erbil within an hour indicates that Δt needs to be shorten.

Another needed improvement for the Lagrangian scheme is in the determination of the vertical distribution of dust at the source site at $t = 0$. Note that the empirical formula for bulk dust fluxes that we used in Chapter 3 and 6 only provide the total fluxes from the surface but not the vertical profile of dust concentration. As such there is ambiguity how the total amount of dust should be distributed to different vertical layers of WRF at $t = 0$. (For the simple demonstration in Fig. 27 (b), we simply put all dust at the lowest level of WRF.) The information about the vertical profile of dust (in relation to turbulence intensity and soil moisture, etc.) could be filled by more complete empirical formulas from future field or lab experiments. Alternatively, one could run a local

advection-diffusion model, using the surface dust fluxes as the boundary condition at the surface, to numerically create the vertical profile. The advective velocity and diffusion coefficient would be deduced from WRF output (from both the resolved velocity and parameterized turbulence variables). However, this is equivalent to locally running an Eulerian transport model (at least in the vertical direction). This points to the potential advantage of simply running an Eulerian transport model in 3-D.

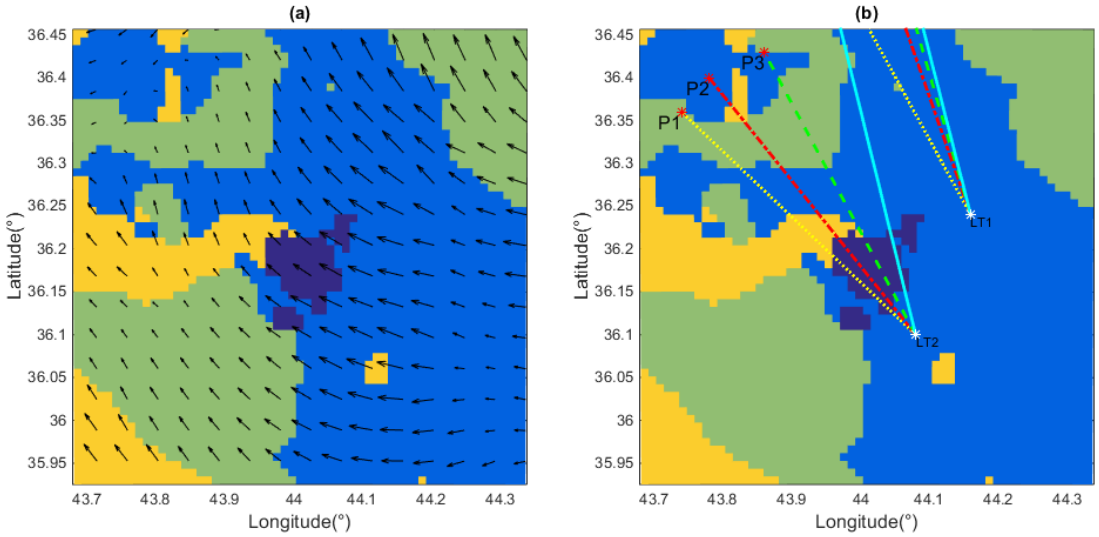

Figure 27 (a) The monthly-mean 10m wind vectors for winter (January). (b)The length and direction of the "1-hour paths" which imply the relevant time scale for the transport calculation. Superimposed in the background is the land-use map for 1987.

(2) *Eulerian transport model*

Given the discussion in the preceding sub-section, an alternative is to run an Eulerian transport model (using the advection-diffusion equation) in 3-D for dust concentration. Note that WRF has a built-in transport model for water vapor and other phases of water by the resolved wind. However, even for the calculation of water budget, the intense exchange of water in the vertical direction (and deposition of liquid or solid water to the surface) due to thermal convection is not merely computed by the transport

equation. Instead, most of it is computed by the physical parameterization schemes for the physical and subgrid-scale processes (for moist convection, cloud physics, and turbulence). For the transport of dust (particularly the initial vertical transport) of our interest, we are facing a similar problem: Over hot desert surfaces, the most efficient way to vertically redistribute the dust from the surface is perhaps by subgrid-scale buoyancy-driven convection. One would then need to build a subgrid-scale parameterization scheme for this process, instead of simply running an advection-diffusion calculation on WRF grids, using resolved wind from WRF output. Completing this task would be of great interest in applications.

In the context of this study, another (relatively minor) point to consider of running an Eulerian transport model is that the information of wind and surface dust fluxes with 1 km horizontal resolution is available only over the innermost domain of WRF. From the quick demonstration in Fig. 27 (b), in the Lagrangian framework most trajectories would leave the innermost domain within one or a few hours, and we anticipate other trajectories to come into the domain from outside. Thus, an issue would arise as to how to set the lateral boundary conditions for the inward and outward dust fluxes. A potential remedy, at the expense of computational time, would be to run WRF with a much larger innermost domain, and perform the Eulerian transport calculation only over a sub-domain of the innermost domain of WRF.

(3) *Deposition of dust*

In the preceding discussions, we assume that the centroid of the dust population is "advected" by the mean wind, in either Lagrangian or Eulerian models. A refinement of the calculation should take into account the general effect of deposition, which depends on the size distribution of dust. Smaller dust particles are passively advected by the mean flow while the movement of larger particles may deviate from that determined by the mean wind, the two scenarios generally distinguished by Stokes number. For the problem of our interest, the mean horizontal velocity of atmospheric wind (~ a few *m/s*) is typically much greater than the mean vertical velocity (~ a few *cm/s*) while the movement of a dust particle is further influenced by gravity in the vertical direction. As such, it is mainly in the vertical direction that one should consider the deviation of the particle velocity from that set by the mean wind. The degree of deviation depends on the terminal velocity of the dust particle, which in term depends on the size of the particle.

Since the empirical formulas for the bulk fluxes of dust used in Chapter 3 and 6 do not provide the size spectrum of dust, we will only outline a conceptual framework of incorporating the effect of (vertical) deposition into the transport scheme. Dust particles are small enough that the air flow passing a falling particle typically has a Reynolds number of $Re < 0.1$ or smaller, and the drag on the particle closely follows Stokes' law. The terminal velocity, w_t, of the particle is determined by the balance between the downward gravity and upward drag force by Stokes formula. For smaller particles (with their sizes approaching the mean free path of air molecules), a minor correction applies by a correction coefficient, $C_c = 1 + (1.257)(2\lambda/d)$ (Allen and Raabe 1982), where λ is the mean free path ($\lambda \sim 6.5 \times 10^{-8}$ m under standard atmospheric conditions) and d is the

diameter of particle. Together, the terminal velocity becomes (Flagan and Seinfeld 1988, McPherson 1993)

$$w_t = \frac{d^2 g(\rho_s - \rho_a) C_c}{18\mu_a} \ , \qquad (18)$$

where g is gravity, μ_a is viscosity of air, and ρ_s and ρ_a are the densities of the dust particle and air. As an example, given ρ_s = 2650 kg/m^3 for clay particles (Nickovic *et al*. 2001), Fig. 28 (a) shows terminal velocity as a function of particle size for 1 μm ≤ d ≤ 10 μm. At d = 10 μm, the terminal velocity is around 1 *cm/s*, comparable to the typical mean vertical velocity of air. If the mean vertical velocity is zero, the particle falls by 30 m in an hour. In contrast, at d = 1 μm the terminal velocity is only 1% of that of a 10-μm particle, and the former simply follows the mean flow velocity. Figure 28 (b) illustrates the particle paths in the *x*-*z* plane as a function of particle size, given the typical horizontal distance travelled by the particle in an hour and an initial height of particles at 30 m. In this case, only large particles will settle onto the city ground while smaller particles are blown away (although they may still affect air quality in the city). Once the information about the spectrum of particles become available, the conceptual framework described here can be readily merged with the transport scheme.

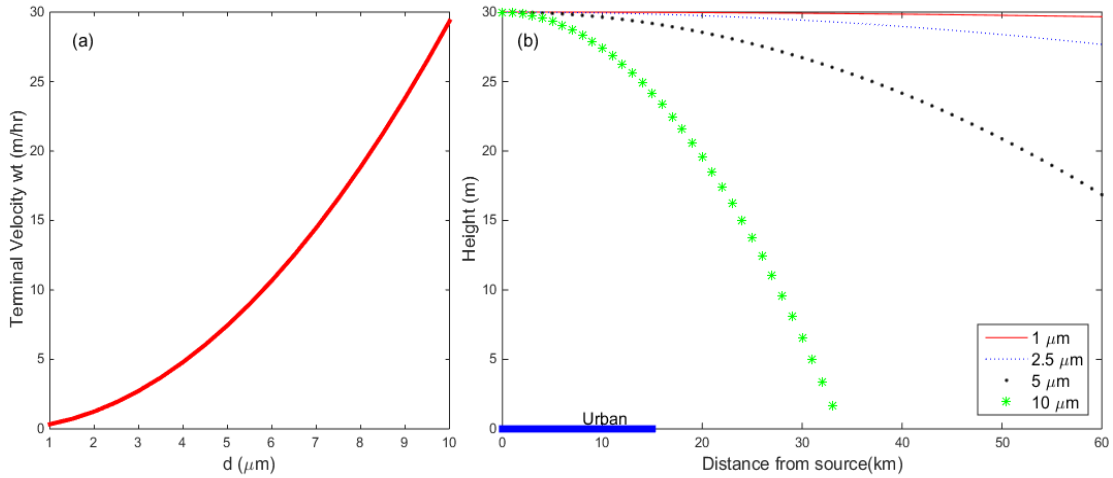

Figure 28: (a) Terminal velocity as a function of the size of a dust particle. (b) Conceptual 2-D paths of particles with different sizes. See text for discussion.

7.2 Improvement in Meteorological Simulations at Urban Scale

Various details in the numerical framework and its input as used in this study can still be improved. In addition to the aspects related to dust transport as detailed in Section 7.1, the more critical future works are summarized in the following.

(1) *Resolving the details at sub-urban scales*

At the 1-km horizontal resolution, our model resolves the basic distribution of the land-use types over the city, but not the detailed "urban topography" associated with street canyons. For a detailed application to local air quality, running the model at a higher resolution to resolve the intra-city landscape will be helpful. Having a further refined grid size will help better resolve not only the airflow, but also the fine detail of the heterogeneous land-surface cover within the city. Having the resolution of street canyons will also make it more realistic to include anthropogenic sources of dust in future numerical simulations.

(2) *Understanding and correcting model biases*

In Chapter 5 we noted that the surface wind in WRF model simulation is potentially biased (to be too strong). While such biases are not uncommon, it is desirable to reduce them to make the model more useful for applications. A simple fix is to perform bias corrections based on comparisons between model and observation, in the fashion of the analysis in Chapter 5 but using a greater number of samples to improve the reliability of statistics. This can be done systematically if the simulations in this study are expanded to multiple years, and using an ensemble of multiple runs. As noted in other studies (e.g., Sharma and Huang 2012), the biases in WRF can be resolution-dependent. Moreover, the biases do not always diminish with an increase in model resolution. In light of our findings, it is useful to further clarify how the high-wind bias depends on resolution and the general meteorological conditions.

(3) *Testing the framework for different cities*

Our framework can, in principle, be applied to other desert cities with different size, baseline climatology (hot vs. cold regions), and dominant land-surface types (e.g., semi-arid land vs. pure desert). Performing more numerical experiments over a range of desert cities will further help us establish the robustness of our results and understand the inter-city differences in terms of the aforementioned local conditions. In particular, we note that the land cover over Erbil and its vicinity is dominated by a small number of land-use categories. It will be useful, particularly for applications, to test our framework on all major land categories associated with urbanization in desert and semi-arid regions. Previously, Kamal et al. (2017) performed an interesting comparison for meso-scale meteorological simulations across five cities in arid and semi-arid regions. An immediate future work will be to extend such a study to include a comparison on dust generation.

CHAPTER 8

CONCLUSION AND RECOMMENDATIONS

8.1 Summary of Key Outcome

This study expands the framework of dynamical downscaling for regional climate modeling to include a component of computation of dust fluxes. The extended framework connects a suite of models, including a meso-scale meteorological model, a land-surface model, an urban canopy model, and a turbulence model, to produce the near-surface turbulence intensity, soil moisture, and land-surface properties. These key parameters are used to determine the surface dust fluxes from laboratory-based empirical formulas. Focusing on natural wind-blown dust, this framework is applied to a set of simulations for the desert city of Erbil in Northern Iraq. The models successfully quantify the differences in the surface dust fluxes generated in the areas surrounding the city between the pre- and post-urbanization eras. The key achievement in this respect is in quantifying the changes in the climatology of surface dust fluxes at the urban scale, under realistic meteorological conditions that fully account for atmosphere-land interaction associated with urbanization.

The output of the numerical models provides the spatial and temporal distributions of surface dust fluxes along 3-D meteorological and land-surface variables. Analyzing these results further helps us understand the dependence of dust production on meteorological conditions and land-surface types. The key achievement in this part of analysis is that we obtained such relations quantitatively, for a wide range of meteorological conditions and different land-use categories. This will allow an easy transfer of our models and results into practical applications for environmental assessment and prediction.

In the process of connecting multiple models to perform numerical simulations for a specific desert city, we also verified and validated the performances of the models in terms of the meteorological fields. Along the way, this helps increase our confidence in the whole modeling system as the horizontal resolution of the meteorological model is pushed to the urban scale. Also, specific aspects of model biases are identified. The knowledge gained from the model verification and validation will help future development of meso-scale meteorological models and future applications of dynamical downscaling, which is essential for conveying the information of global climate change to regional and urban scales.

8.2 Recommendations

The multi-model framework used and tested in this study can be applied to other desert cities, and cities in semi-arid regions, where generation of wind-blown dust is of primary importance to the local environment. Performing such calculations globally, one can obtain the critical information of the changes in the concentration of airborne dust under future scenarios of global climate changes and land-use changes. This has implications for not only air quality but also the radiative effect of dust on the energy balance of global atmosphere.

In local applications, the framework developed in this study can be connected to tools used by local stakeholders for the assessment and control of air quality. For a more efficient use of the framework in this context, one can further synthesize the numerical simulations into empirical relations or look-up tables that connect local meteorological conditions to local dust fluxes. In this context, we conclude this study with a few examples of potential applications:

(i) *Using the outcome of numerical models to assist urban planning*: Our models can be used to predict future distribution of surface dust fluxes and meteorological variables (wind, temperature, precipitation) associated with different scenarios of urban expansion. This allows local stakeholders to quantitatively assess the environmental impacts of different, and potentially competing, plans for urban development.

(ii) *Using the models to assist the monitoring and regulation of air quality*: The detailed relation between meteorological conditions and surface dust fluxes on different types of land surfaces can be used to create a warning system. For example, with an unusually high wind speed that exceeds a certain threshold, a warning can be triggered to alert to a

potentially harmful dusty condition. The system can be integrated into a more comprehensive monitoring system for public health.

(iii) *Industrial applications*: Many industries, particularly in the high-precision manufacturing sector, require low-dust working conditions. For example, manufacturing of micro-scale semiconductor components, medical equipment, and pharmaceutical products require an environment with very low dust concentration. The information produced by the numerical models can help with the selection of sites for the factories for those industries.

(iv)*Value-added weather prediction*: The existing system for routine daily weather forecast provides mainly the outlook of meteorological variables (wind, temperature, precipitation). With the relation between those variables and dust fluxes obtained from this study, an add-on component can be developed to extend regular weather forecast to one that includes some information on potentially dusty conditions at the urban scale.

REFERENCES

Bigelow, D. P., & Borchers, A. (2017). Major Uses of Land in the United States, 2012EIB-178. US: U.S. Department of Agriculture, Economic Research Service.

Blackadar, A. K. (1997). Turbulence and Diffusion in the Atmosphere. New York: Springer.

Chen, F. Kusaka, H. Borndtein, J. Ching, J. (2011). The integrated WRF/urban modelling system: development, evaluation, and applications to urban environmental problems. International Journal of Climatology, DOI: 10.1002/joc.2158, 273-288.

Choi, Y.-J., & Fernando, H. (2008). Implementation of a windblown dust parameterization into MODELS-3/CMAQ: Application to episodic PM events in the US/Mexico border. Atmospheric Environment, 6039– 6046.

Fecan, F., Marticorena, B., Bergametti, G., 1999. Parameterization of the increase of the Aeolian erosion threshold wind friction velocity due to soil moisture for arid and semi-arid areas. Annales de Geophysique 17, 149–157.

Faqe Ibrahim, G., (2015). Urban Expansion Monitoring Utilizing Remote Sensing tools in Erbil City- Kurdistan Region. Master Thesis. Soran University, Erbil, Iraq

Flagan, Richard C., and John H. Seinfeld. 1988. Fundamentals of Air Pollution Engineering. New Jersey: Prentice-Hall, Inc.

Gong, S.L., Zhang, X.Y., Zhao, T.L., Mckendry, I.G., Jaffe, D.A., Lu, N.M., 2003. Characterization of soil dust aerosol in China and its transport and distribution during 2001 ACE-Asia:2. Model simulation and validation. Journal of Geophysical Research. 108 doi:10.1029/2002JD002633

Hong, S.-Y., & Pan, H.-L. (1995). Nonlocal Boundary Layer Vertical Diffusion in a Medium-Range Forecast Model. Monthly Weather Review, 124, 2322-2339.

Hong, S.-Y., & Noh, Y. (2005). A New Vertical Diffusion Package with an Explicit Treatment of Entrainment Processes. Monthly Weather Review, 134, 2318-2341

Huneeus, N., Schulz, M., Balkanski, Y., & Griesfeller, J. (2011). Global dust model inter-comparison in AeroCom phase I. Atmospheric Chemistry and Pgysics, 7782-7816.

Kusaka, H., H. Kondo, Y. Kikegawa, and F. Kimura, 2001: A simple single-layer urban canopy model for atmospheric models: Comparison with multi-layer and slab models. Bound.-Layer Meteor., 101, 329–358.

Kusaka, H., and F. Kimura, 2004: Coupling a single-layer urban canopy model with a simple atmospheric model: Impact on urban heat island simulation for an idealized case, J. Meteor. Soc. Japan, 82, 67-80

Kamal, S. (2015). Numerical Study of the Effect of Urbanization on the Climate of Desert Cities. Doctoral dissertation. Arizona State University, Tempe, Arizona (a)

Kamal, S. Huang, H. Myint, S. (2015). The Influence of Urbanization on the Climate of the Las Vegas Metropolitan Area: A Numerical Study. Journal of Applied Meteorology and Climatology, V54, DOI: 10.1175/JAMC-D-15-0003.1(b)

Kamal, S. Huang, H. Myint, S. (2017). Numerical simulations to quantify the diurnal contrast in local climate trend induced by desert urbanization. Environment Systems and Decisions, DOI 10.1007/s10669-017-9657-2, pp13.

Kok, J. F. (2011). Does the size distribution of mineral dust aerosols depend on the wind speed at mission? Atmospheric Chemistry and Physics, 10149-10156.

Kok, J. F., & Mahowald, N. M. (2014). An improved dust emission model – Part 1: Model description and comparison against measurements. Atmospheric Chemistry and Physics, 13023- 3041.

Kok, J. F., Albani, S., & Mahowald, N. M. (2014). An improved dust emission model – Part 2: Evaluation in the Community Earth System Model, with implications for the use of dust source functions. Atmospheric Chemistry and Physics, 13043-13061.

Laurent, B., Marticorena, B., & Bergametti, G. (2009). Modelling mineral dust emissions. Earth and Environmental Science. doi:10.1088/1755-1307/7/1/012006

Ma, X., Bafrtlett, K., Harmon, K., & Yu, F. (2013). Comparison of AOD between CALIPSO and MODIS: significant differences over major dust and biomass burning regions. Atmospheric Measurement Techniques, 2391-2401.

McDonald, R. I., Marcotullio, P. J., & Güneralp, B. (2013). Urbanization and Global Trends in Biodiversity and Ecosystem Services. In C. N. Norbert Müller, Pattern and Trends in Urban Biodiversity and Landscape Design (pp. 31-52). New York: Wayne C. Zipperer.

McPherson, Malcolm J. (1993). "The aerodynamics, sources and control of airborne dust." In Subsurface Ventilation and Environmental Engineering, by Malcolm J. McPherson, 1-40. New Delhi, Thomson Press (India) Limited.

Mesoscale & Microscale Meteorology Division. (2016). WRF, ARW Version 3 Modeling System User's Guide, National Center for Atmospheric Research.

Nickovic, S., Kallos, G., Papadopoulos, A., Kakaliagou, O., 2001. A model for prediction of desert dust cycle in the atmosphere. Journal of Geophysical Research 106, 18113–18129.

Rasul, A. (2015). Spatial variation of the daytime Surface Urban Cool Island during the dry season in Erbil, Iraqi Kurdistan, from Landsat 8. Urban Climate, 176-186.

Rasul, A. (2016). Remote Sensing of Surface Urban Cool and Heat Island Dynamics in Erbil, Iraq, between 1992 and 2013. Doctoral dissertation, University of Leicester.

Saha, S., et al. 2010. NCEP Climate Forecast System Reanalysis (CFSR) 6-hourly Products, January 1979 to December 2010. Research Data Archive at the National Center for Atmospheric Research, Computational and Information Systems Laboratory. https://doi.org/10.5065/D69K487J. Accessed† 12 Dec. 2018.

Saha, S., Moorthi, S., Wu, X., Wang, J., Nadiga, S. 2014. "The NCEP Climate Forecast System Version 2." Journal of Climate

Sharma A, Huang, H-P. (2012). "Regional Climate Simulation for Arizona: Impact of Resolution on Precipitation." Advances in Meteorology. doi:10.1155/2012/505726

Schepanski, K. (2014). Mineral dust: Meteorological controls and climate impacts. PAGES MAGAZINE, 62-63.

Skamarock, W. C., Klemp, J. B., Dudhia, J., Gill, D. O., Barker, D. M., Duda, M. G., Powers, J. G. (2008). A Description of the Advanced Research WRF Version 3. Boulder, Colorado: National Center for Atmospheric Research.

Shigeta, Y., Yukitaka O., and Tsukamoto O. 2009. "Urban Cool Island in Daytime －Analysis by Using Thermal Image and Air Temperature Measurements－." The seventh International Conference on Urban Climate. Yokohama..

Stein, A., Draxler, R., Rolph, G., Stunder, B., Cohen, M., & Ngan, F. (2015). NOAA'S HYSPLIT Atmospheric Transport and Dispersion Modeling System. American Meteorological Society.

Tegen, I., and I. Fung, Modeling of mineral dust in the atmosphere: Sources, transport, and optical thickness. Journal of Geophysical Research, Vol. 99, No. D11, Pages 22,987-22,914, 1994.

United Nation, Department of Economic and Social Affairs, Population Division (2017). (2017). World Population Prospects 2017 – Data Booklet. New York: United Nation.

United Nations, Department of Economic and Social Affairs, Population Division. (2017). World Population Prospects: The 2017 Revision, Key Findings and Advance Tables. New York: United Nations, Department of Economic and Social Affairs.

Westphal, D. Toon, O. Carlson, T (1987). A Two-Dimensional Numerical Investigation of the Dynamics and Microphysics of Saharan Dust Storms. Journal of Geophysical Research, 92, 3027-3049.

APPENDIX A

WRF INPUT FILES

EXAMPLES OF THE INPUT DATA USED FOR WRF MODEL SIMULATION

I. Namelist.wps

```
&share
 wrf_core             = 'ARW',
 max_dom              = 3,
 start_date = '2000-07-01_00:00:00', '2000-07-01_00:00:00', '2000-07-01_00:00:00',
 end_date   = '2000-07-31_00:00:00', '2000-07-31_00:00:00', '2000-07-31_00:00:00',
 interval_seconds     = 10800,
 io_form_geogrid      = 2,
 debug_level          = 0,
 /
&geogrid
 parent_id            = 1,1,2,
 parent_grid_ratio    = 1,5,5,
 i_parent_start       = 1,8,15,
 j_parent_start       = 1,8,15,
 e_we                 = 20,41,61,
 e_sn                 = 20,41,61,
 geog_data_res        = '10m','5m','2m',
 dx                   = 25000,
 dy                   = 25000,
 map_proj             =  'lambert',
 ref_lat              = 36.1911,
 ref_lon              = 44.0091,
 truelat1             = 36,
 truelat2             = 36,
 stand_lon            = 44,
 geog_data_path    = '/media/usr3/stahir/Erbil2000SF/Erbil2011SF/geog',
 ref_x                = 11.5,
 ref_y                = 11.5,
/
&ungrib
 out_format          = 'WPS',
 prefix              = 'FILE',
/
&metgrid
 fg_name             = 'FILE'
 io_form_metgrid = 2,
/
```

II. Namelist.input

```
&time_control
 run_days                  = 30,
 run_hours                 = 00,
 run_minutes               = 0,
 run_seconds               = 0,
 start_year                = 2000, 2000, 2000,
 start_month               = 07,   07,   07,
 start_day                 = 01,   01,   01,
 start_hour                = 00,   00,   00,
 start_minute              = 00,   00,   00,
 start_second              = 00,   00,   00,
 end_year                  = 2000, 2000, 2000,
 end_month                 = 07,   07,   07,
 end_day                   = 31,   31,   31,
 end_hour                  = 00,   00,   00,
 end_minute                = 00,   00,   00,
 end_second                = 00,   00,   00,
 interval_seconds          = 10800
 input_from_file           = .true.,.true.,.true.,
 history_interval          = 180,  180,   180,
 frames_per_outfile        = 1000, 1000, 1000,
 restart                   = .false.,
 restart_interval          = 5000,
 io_form_history           = 2
 io_form_restart           = 2
 io_form_input             = 2
 io_form_boundary          = 2
 debug_level               = 0
/
&domains
 time_step                 = 180,
 time_step_fract_num       = 0,
 time_step_fract_den       = 1,
 max_dom                   = 3,
 e_we                      = 20,    41,    61,
 e_sn                      = 20,    41,    61,
 e_vert                    = 28,    28,    28,
 p_top_requested           = 5000,
 num_metgrid_levels        = 27
 num_metgrid_soil_levels   = 2,
 dx                        = 25000, 5000, 1000,
 dy                        = 25000, 5000, 1000,
 grid_id                   = 1,     2,    3,
 parent_id                 = 1,     1,    2,
```

```
i_parent_start                      = 1,     8,    15,
j_parent_start                      = 1,     8,    15,
parent_grid_ratio                   = 1,     5,    5,
parent_time_step_ratio              = 1,     5,    5,
feedback                            = 1,
smooth_option                       = 0
/
&physics
mp_physics                          = 3,     3,     3,
ra_lw_physics                       = 1,     1,     1,
ra_sw_physics                       = 1,     1,     1,
radt                                = 30,    30,    30,
sf_sfclay_physics                   = 1,     1,     1,
sf_surface_physics                  = 2,     2,     2,
bl_pbl_physics                      = 1,     1,     1,
bldt                                = 0,     0,     0,
cu_physics                          = 1,     1,     0,
cudt                                = 5,     5,     5,
isfflx                              = 1,
ifsnow                              = 0,
icloud                              = 1,
surface_input_source                = 1,
num_soil_layers                     = 4,
sf_urban_physics                    = 0,     0,     1,
/
&fdda
/
&dynamics
w_damping                           = 0,
diff_opt                            = 1,
km_opt                              = 4,
diff_6th_opt                        = 0,      0,      0,
diff_6th_factor                     = 0.12,   0.12,   0.12,
base_temp                           = 290.
damp_opt                            = 0,
zdamp                               = 5000.,  5000.,  5000.,
dampcoef                            = 0.2,    0.2,    0.2
khdif                               = 0,      0,      0,
kvdif                               = 0,      0,      0,
non_hydrostatic                     = .true., .true., .true.,
moist_adv_opt                       = 1,      1,      1,
scalar_adv_opt                      = 1,      1,      1,
/
&bdy_control
spec_bdy_width                      = 5,
spec_zone                           = 1,
```

```
relax_zone                = 4,
specified                 = .true., .false.,.false.,
nested                    = .false., .true., .true.,
/
&grib2
/
&namelist_quilt
nio_tasks_per_group       = 0,
nio_groups                = 1,
/
```

III. NOAA FNL Data Set Used for the Lateral Boundary condition

```
#!/bin/csh
#################################################################################
# Csh Script to retrieve 124 online Data files of 'ds083.2',
# total 2.58G. This script uses 'wget' to download data.
#
# Highlight this script by Select All, Copy and Paste it into a file;
# make the file executable and run it on command line.
#
# You need pass in your password as a parameter to execute
# this script; or you can set an environment variable RDAPSWD
# if your Operating System supports it.
#
# Contact grace@ucar.edu (Grace Peng) for further assistance.
#################################################################################
set pswd     = $1
if(x$pswd == x && `env | grep RDAPSWD` != '') then
set pswd    = $RDAPSWD
endif
if(x$pswd == x) then
echo
echo Usage: $0 YourPassword
echo
exit 1
endif
set v         = `wget -V |grep 'GNU Wget ' | cut -d ' ' -f 3`
set a         = `echo $v | cut -d '.' -f 1`
set b         = `echo $v | cut -d '.' -f 2`
if(100 * $a + $b > 109) then
set opt       = 'wget --no-check-certificate'
else
set opt       = 'wget'
endif
set opt1     = '-O Authentication.log --save-cookies auth.rda_ucar_edu --post-data'
set opt2     = "email=sttahir@asu.edu&passwd=$pswd&action=login"
$opt $opt1="$opt2" https://rda.ucar.edu/cgi-bin/login
set opt1    = "-N --load-cookies auth.rda_ucar_edu"
set opt2    = "$opt $opt1 http://rda.ucar.edu/data/ds083.2/"
set filelist = ( \
grib1/2000/2000.07/fnl_20000701_00_00.grib1 \
grib1/2000/2000.07/fnl_20000701_06_00.grib1 \
grib1/2000/2000.07/fnl_20000701_12_00.grib1 \
grib1/2000/2000.07/fnl_20000701_18_00.grib1 \
grib1/2000/2000.07/fnl_20000702_00_00.grib1 \
grib1/2000/2000.07/fnl_20000702_06_00.grib1 \
```

grib1/2000/2000.07/fnl_20000702_12_00.grib1 \
grib1/2000/2000.07/fnl_20000702_18_00.grib1 \
grib1/2000/2000.07/fnl_20000703_00_00.grib1 \
grib1/2000/2000.07/fnl_20000703_06_00.grib1 \
grib1/2000/2000.07/fnl_20000703_12_00.grib1 \
grib1/2000/2000.07/fnl_20000703_18_00.grib1 \
grib1/2000/2000.07/fnl_20000704_00_00.grib1 \
grib1/2000/2000.07/fnl_20000704_06_00.grib1 \
grib1/2000/2000.07/fnl_20000704_12_00.grib1 \
grib1/2000/2000.07/fnl_20000704_18_00.grib1 \
grib1/2000/2000.07/fnl_20000705_00_00.grib1 \
grib1/2000/2000.07/fnl_20000705_06_00.grib1 \
grib1/2000/2000.07/fnl_20000705_12_00.grib1 \
grib1/2000/2000.07/fnl_20000705_18_00.grib1 \
grib1/2000/2000.07/fnl_20000706_00_00.grib1 \
grib1/2000/2000.07/fnl_20000706_06_00.grib1 \
grib1/2000/2000.07/fnl_20000706_12_00.grib1 \
grib1/2000/2000.07/fnl_20000706_18_00.grib1 \
grib1/2000/2000.07/fnl_20000707_00_00.grib1 \
grib1/2000/2000.07/fnl_20000707_06_00.grib1 \
grib1/2000/2000.07/fnl_20000707_12_00.grib1 \
grib1/2000/2000.07/fnl_20000707_18_00.grib1 \
grib1/2000/2000.07/fnl_20000708_00_00.grib1 \
grib1/2000/2000.07/fnl_20000708_06_00.grib1 \
grib1/2000/2000.07/fnl_20000708_12_00.grib1 \
grib1/2000/2000.07/fnl_20000708_18_00.grib1 \
grib1/2000/2000.07/fnl_20000709_00_00.grib1 \
grib1/2000/2000.07/fnl_20000709_06_00.grib1 \
grib1/2000/2000.07/fnl_20000709_12_00.grib1 \
grib1/2000/2000.07/fnl_20000709_18_00.grib1 \
grib1/2000/2000.07/fnl_20000710_00_00.grib1 \
grib1/2000/2000.07/fnl_20000710_06_00.grib1 \
grib1/2000/2000.07/fnl_20000710_12_00.grib1 \
grib1/2000/2000.07/fnl_20000710_18_00.grib1 \
grib1/2000/2000.07/fnl_20000711_00_00.grib1 \
grib1/2000/2000.07/fnl_20000711_06_00.grib1 \
grib1/2000/2000.07/fnl_20000711_12_00.grib1 \
grib1/2000/2000.07/fnl_20000711_18_00.grib1 \
grib1/2000/2000.07/fnl_20000712_00_00.grib1 \
grib1/2000/2000.07/fnl_20000712_06_00.grib1 \
grib1/2000/2000.07/fnl_20000712_12_00.grib1 \
grib1/2000/2000.07/fnl_20000712_18_00.grib1 \
grib1/2000/2000.07/fnl_20000713_00_00.grib1 \
grib1/2000/2000.07/fnl_20000713_06_00.grib1 \
grib1/2000/2000.07/fnl_20000713_12_00.grib1 \
grib1/2000/2000.07/fnl_20000713_18_00.grib1 \

grib1/2000/2000.07/fnl_20000714_00_00.grib1 \
grib1/2000/2000.07/fnl_20000714_06_00.grib1 \
grib1/2000/2000.07/fnl_20000714_12_00.grib1 \
grib1/2000/2000.07/fnl_20000714_18_00.grib1 \
grib1/2000/2000.07/fnl_20000715_00_00.grib1 \
grib1/2000/2000.07/fnl_20000715_06_00.grib1 \
grib1/2000/2000.07/fnl_20000715_12_00.grib1 \
grib1/2000/2000.07/fnl_20000715_18_00.grib1 \
grib1/2000/2000.07/fnl_20000716_00_00.grib1 \
grib1/2000/2000.07/fnl_20000716_06_00.grib1 \
grib1/2000/2000.07/fnl_20000716_12_00.grib1 \
grib1/2000/2000.07/fnl_20000716_18_00.grib1 \
grib1/2000/2000.07/fnl_20000717_00_00.grib1 \
grib1/2000/2000.07/fnl_20000717_06_00.grib1 \
grib1/2000/2000.07/fnl_20000717_12_00.grib1 \
grib1/2000/2000.07/fnl_20000717_18_00.grib1 \
grib1/2000/2000.07/fnl_20000718_00_00.grib1 \
grib1/2000/2000.07/fnl_20000718_06_00.grib1 \
grib1/2000/2000.07/fnl_20000718_12_00.grib1 \
grib1/2000/2000.07/fnl_20000718_18_00.grib1 \
grib1/2000/2000.07/fnl_20000719_00_00.grib1 \
grib1/2000/2000.07/fnl_20000719_06_00.grib1 \
grib1/2000/2000.07/fnl_20000719_12_00.grib1 \
grib1/2000/2000.07/fnl_20000719_18_00.grib1 \
grib1/2000/2000.07/fnl_20000720_00_00.grib1 \
grib1/2000/2000.07/fnl_20000720_06_00.grib1 \
grib1/2000/2000.07/fnl_20000720_12_00.grib1 \
grib1/2000/2000.07/fnl_20000720_18_00.grib1 \
grib1/2000/2000.07/fnl_20000721_00_00.grib1 \
grib1/2000/2000.07/fnl_20000721_06_00.grib1 \
grib1/2000/2000.07/fnl_20000721_12_00.grib1 \
grib1/2000/2000.07/fnl_20000721_18_00.grib1 \
grib1/2000/2000.07/fnl_20000722_00_00.grib1 \
grib1/2000/2000.07/fnl_20000722_06_00.grib1 \
grib1/2000/2000.07/fnl_20000722_12_00.grib1 \
grib1/2000/2000.07/fnl_20000722_18_00.grib1 \
grib1/2000/2000.07/fnl_20000723_00_00.grib1 \
grib1/2000/2000.07/fnl_20000723_06_00.grib1 \
grib1/2000/2000.07/fnl_20000723_12_00.grib1 \
grib1/2000/2000.07/fnl_20000723_18_00.grib1 \
grib1/2000/2000.07/fnl_20000724_00_00.grib1 \
grib1/2000/2000.07/fnl_20000724_06_00.grib1 \
grib1/2000/2000.07/fnl_20000724_12_00.grib1 \
grib1/2000/2000.07/fnl_20000724_18_00.grib1 \
grib1/2000/2000.07/fnl_20000725_00_00.grib1 \
grib1/2000/2000.07/fnl_20000725_06_00.grib1 \

```
grib1/2000/2000.07/fnl_20000725_12_00.grib1 \
grib1/2000/2000.07/fnl_20000725_18_00.grib1 \
grib1/2000/2000.07/fnl_20000726_00_00.grib1 \
grib1/2000/2000.07/fnl_20000726_06_00.grib1 \
grib1/2000/2000.07/fnl_20000726_12_00.grib1 \
grib1/2000/2000.07/fnl_20000726_18_00.grib1 \
grib1/2000/2000.07/fnl_20000727_00_00.grib1 \
grib1/2000/2000.07/fnl_20000727_06_00.grib1 \
grib1/2000/2000.07/fnl_20000727_12_00.grib1 \
grib1/2000/2000.07/fnl_20000727_18_00.grib1 \
grib1/2000/2000.07/fnl_20000728_00_00.grib1 \
grib1/2000/2000.07/fnl_20000728_06_00.grib1 \
grib1/2000/2000.07/fnl_20000728_12_00.grib1 \
grib1/2000/2000.07/fnl_20000728_18_00.grib1 \
grib1/2000/2000.07/fnl_20000729_00_00.grib1 \
grib1/2000/2000.07/fnl_20000729_06_00.grib1 \
grib1/2000/2000.07/fnl_20000729_12_00.grib1 \
grib1/2000/2000.07/fnl_20000729_18_00.grib1 \
grib1/2000/2000.07/fnl_20000730_00_00.grib1 \
grib1/2000/2000.07/fnl_20000730_06_00.grib1 \
grib1/2000/2000.07/fnl_20000730_12_00.grib1 \
grib1/2000/2000.07/fnl_20000730_18_00.grib1 \
grib1/2000/2000.07/fnl_20000731_00_00.grib1 \
grib1/2000/2000.07/fnl_20000731_06_00.grib1 \
grib1/2000/2000.07/fnl_20000731_12_00.grib1 \
grib1/2000/2000.07/fnl_20000731_18_00.grib1 \
)
while($#filelist > 0)
set syscmd = "$opt2$filelist[1]"
echo "$syscmd ..."
$syscmd
shift filelist
end
rm -f auth.rda_ucar_edu Authentication.log
exit 0
```

APPENDIX B

MATLAB CODES

MATLAB CODES FOR PREPROCESSING AND POST-PROCESSING OF WRF SIMULATIONS

I. Facilitating Changes in Land-Surface Cover in the Surface Boundary Condition

A. Changing the Land Cover from Non-Urban to Urban types

```
clc
clear all
%%opening the netcdf files
ncid2 = netcdf.open('Test01.nc','NC_WRITE');
lat2 = double(netcdf.getVar(ncid2,1,[0 0 0],[60 60 1]));
long2 = double(netcdf.getVar(ncid2,2,[0 0 0],[60 60 1]));
%%getting the Lu_index(variable number 24)
L2 = double(netcdf.getVar(ncid2,24,[0 0 0],[60 60 1]));
%%getting the Landusefraction variable number 23
Lf2 = double(netcdf.getVar(ncid2,23,[0 0 0 0],[60 60 24 1]));
%%reshaping all the arrays to get a rid of the time dim
L2=(squeeze(L2(:,:,1)));
Lf2=(squeeze(Lf2(:,:,:,1)));
long2= double(squeeze(long2(:,:,1)));
lat2= double(squeeze(lat2(:,:,1)));
L2n=L2;Lf2n=Lf2;
quick1 = 0;
for i=1:60;   for j=1:60;
%% Made a box with the corners at (43.9, 36.1), (44.1, 36.3) and checked for the grid
cells
%% Find where the LU_INDEX doesn't correspond to a desert, shrub land or water (the
only thing left is urban build up).It is important to know
%% that this choice is flexible and can be modified to better capture the physical
boundaries of a city.
If ((long2(I,j)> 43.97) && (long2(I,j)<43.98) && (lat2(I,j) > 36.08) && (lat2(I,j)
<36.11)&& L2(I,j)==6 );
%% Turning the LU_index to urban
L2n(I,j)=1;
%% At an urban grid point, turn the array of LUF to (0.9,0,0,0,0,0,0,0.1,0,0,...)
for k = 1:24
Lf2n(I,j,k)=0;
end
Lf2n(I,j,1)=0.9; Lf2n(I,j,8)=0.1;
quick1 = quick1+1;
end;   end; end
quick1
%%Replace the old LU_INDEX and LANDUSEF arrays by my new modified values
netcdf.putVar(ncid2,24,L2n);netcdf.putVar(ncid2,23,Lf2n);
```

B. Validating the Modification of Land Surface Boundary Condition

```
clc
clear all
p=60; % x-axiz length
q=60; % y-axis length
% Open default (Geo LULC) of WRF file for Erbil
ncdisp('geo_em.d03.nc');
ncid6 = netcdf.open('geo_em.d03.nc','NC_NOWRITE');
%Reading Latitude and longitude coordinates
long6 = netcdf.getVar(ncid6,2,[0 0 0],[p q 1]);
lat6 = netcdf.getVar(ncid6,1,[0 0 0],[p q 1]);
% Reading Land Mask Index
lu6 = netcdf.getVar(ncid6,24,[0 0 0],[p q 1]);
latd6 = double(lat6);
longd6 = double(long6);
lud6 = double(lu6);
hold on
% Plotting the WRF default LULC map for Erbil
figure (1)
ax1 = subplot(2,1,1,'Position',[0.29,0.55,0.4,0.4]);
h1=pcolor(longd6,latd6,lud6);
colorbar('Ticks',[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24],...
   'TickLabels',{'1', '2','3','4','5','6','7','8','9'},'FontSize',12);
caxis([1 9]);
title('(a)WRF Default', 'Color', 'k');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
pbaspect([1 1 1])
fig = gcf;
fig.PaperPositionMode = 'auto'
% --------------------------------
% Open converted to 1987 file for Erbil
ncdisp('Geo1987Final.nc');
ncid4 = netcdf.open('Geo1987Final.nc','NC_NOWRITE');
%Latitude and longitude coordinates
long4 = netcdf.getVar(ncid4,2,[0 0 0],[p q 1]);
lat4 = netcdf.getVar(ncid4,1,[0 0 0],[p q 1]);
% Land Mask Index
lu4 = netcdf.getVar(ncid4,24,[0 0 0],[p q 1]);
latd4 = double(lat4);
longd4 = double(long4);
lud4 = double(lu4);
hold on
% Plot LULC map for Erbil 1987
figure (1)
ax2 = subplot(2,2,3,'Position',[0.14,0.065,0.4,0.4]);
```

```
h1=pcolor(longd4,latd4,lud4);
caxis([1 9]);
title('(b)1987','Color', 'k');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
fig = gcf;
fig.PaperPositionMode = 'auto'
pbaspect([1 1 1])
%----------------------------------------
% Map of Erbil 2011
ncid2 = netcdf.open('Geo2011Final.nc','NC_NOWRITE');
%Latitude and longitude coordinates
long1 = netcdf.getVar(ncid2,2,[0 0 0],[p q 1]);
lat1 = netcdf.getVar(ncid2,1,[0 0 0],[p q 1]);
% Land Mask Index
lu1 = netcdf.getVar(ncid2,24,[0 0 0],[p q 1]);
latd1 = double(lat1);
longd1 = double(long1);
lud1 = double(lu1);
hold on
% Plot LULC map for Erbl 2011
figure (1)
ax3 = subplot(2,2,4,'Position',[0.41,0.065,0.4,0.4]);
fig = gcf;
fig.PaperPositionMode = 'auto'
pbaspect([1 1 1])
h1=pcolor(longd1,latd1,lud1);
caxis([1 9]);
title('(c)2011','Color', 'k');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
fig = gcf;
fig.PaperPositionMode = 'auto'
pbaspect([1 1 1])
```

II. Analyzing the Meteorological Fields from WRF Output

A. Analyzing Surface Air Temperature

```
clc
clear all
close all
n=240; %% n is time dim length
p=60; % x-axiz length
q=60; % y-axis length
ncdisp('WRFOutSF_W1987.nc');
ncdisp('WRFOutSF_W2011.nc');
ncid1 = netcdf.open('WRFOutSF_W1987.nc','NC_NOWRITE');
ncid2 = netcdf.open('WRFOutSF_W2011.nc','NC_NOWRITE');
%Temperature Reading:
T87W = netcdf.getVar(ncid1,33,[0 0 0],[p q n]);
T11W = netcdf.getVar(ncid2,33,[0 0 0],[p q n]);
Td87W = double(T87W); Td11 = double(T11W);
%Load required maps for winter:
load('Uw87s1.mat','Uw87s1');% Threshold U* winter 1987
load ('R0_1987.mat','R0_1987'); % R Factor map 1987
load('Uw11s1.mat','Uw11s1');% Threshold U* winter 2011
load ('R0_2011.mat','R0_2011'); % R Factor map 2011
load land1987; % Land use map 1987
load land2011; % Land use map 2011
% Reading the 2m Temperature on the newly developed urban region
============
for k = 1:n
   Tmax87W(k) = 0;
   Tmax11W(k) = 0;
   for i = 12:40
   for j = 12:40
      if (lud1(i,j) == 1);
        landmask1(i,j) = 1;
         Tmax87W(k) = Td87W(i,j,k);
         Tmax11W(k) = Td11(i,j,k);
    end
   end
   end
   Tmax87W(k);
   Tmax11W(k);
   timestamp(k) = k/8;
   end
figure (1)
ax1W = subplot(1,2,1,'Position',[0.1,0.3,0.39,0.39]);
plot(timestamp,Tmax87W,'k',timestamp,Tmax11W,'r','LineWidth',2);
title('(a)Winter, 2000','Color','k');
lgd=legend('1987','2011'); lgd.FontSize = 12;
```

```
      xlabel('Time(Day)','FontSize',12);ylabel('Temperature(°K)','FontSize',12);
      ylim([270 295]);
%---------------------------
%% open a netcdf file for summer%%
ncdisp('WRFOutSF_S1987.nc');
ncdisp('WRFOutSF_S2011.nc');
ncid3 = netcdf.open('WRFOutSF_S1987.nc','NC_NOWRITE');
ncid4 = netcdf.open('WRFOutSF_S2011.nc','NC_NOWRITE');
T87S = netcdf.getVar(ncid3,33,[0 0 0],[p q n]);
T11S = netcdf.getVar(ncid4,33,[0 0 0],[p q n]);
Td87S = double(T87S); Td11S = double(T11S);
% Reading the 2m Temperature on the newly developed urban region ============
for k = 1:n
   Tmax87S(k) = 0;
   Tmax11S(k) = 0;
   for i = 12:40
   for j = 12:40
      if (lud4(i,j) == 1);
        landmask4(i,j) = 1;
      else;
        landmask4(i,j) = 0;
      end
      if (lud1(i,j) == 1);
        landmask1(i,j) = 1;
         Tmax87S(k) = Td87S(i,j,k);
         Tmax11S(k) = Td11S(i,j,k);
      else;
        landmask1(i,j) = 0;
     end
   end
   end
   Tmax87S(k);
   Tmax11S(k);
   timestamp(k) = k/8;
   end
figure (1)
ax1 = subplot(1,2,2,'Position',[0.53,0.3,0.39,0.39]);
plot(timestamp,Tmax87S,'k',timestamp,Tmax11S,'r','LineWidth',2);
title('(b)Summer, 2000','Color','k','FontSize',12);
lgd=legend('1987','2011'); lgd.FontSize = 12;
xlabel('Time (Day)','FontSize',12);ylabel('Temperature(°K)','FontSize',12);
ylim([300 330]);
```

B. Comparing 2 m Temperature from WRF Out to CFSR Data

```
clc
clear all
% Reading data from excel sheet and plot them for L1
num1 = xlsread('T2_L1_Compare.xlsx','All Readings');
Day = num1(:,1);
% Comparing daily 2m Temperature during winter at L1
T87 = num1(:,2);
T_L1 = num1(:,4);
 figure (1)
ax1 = subplot(2,2,1,'Position',[0.1,0.57,0.39,0.39]);
plot(Day,T87,'k',Day,T_L1,'r','LineWidth',2);
title('(a)2m Temperature Comparison at L1, winter', 'Color', 'k');
lgd = legend('T87: Temp 1987','T-L1: CFSR Temperature at L1');
lgd.FontSize = 12;
xlabel('Time (Day)','FontSize',12);ylabel('Temp (K°)','FontSize',12);
ylim([270 295]);
pbaspect([2.2 1 1]);
% Reading data from excel sheet and plot them for L1
num1 = xlsread('T2_L1_Compare.xlsx','All reading Ave');
Day = num1(:,1);
% Comparing average 2m Temperature during winter at L1
T87 = num1(:,2);
T_L1 = num1(:,4);
 figure (1)
ax1 = subplot(2,2,3,'Position',[0.1,0.13,0.39,0.39]);
plot(Day,T87,'k',Day,T_L1,'r','LineWidth',2);
title('(b)Average 2m Temperature Comparison at L1, winter','Color','k');
lgd = legend('T87: Average Temp.1987','T-L1: CFSR Temp. at L1');
lgd.FontSize = 12;
xlabel('Time (Day) ','FontSize',12);ylabel('Temp (K°)','FontSize',12);
ylim([270 295]);
pbaspect([2.2 1 1]);
%-----------------------------------------------
% Reading data from excel sheet and plot them for L2
num1 = xlsread('T2_L2_Compare.xlsx','All Readings');
Day = num1(:,1);
% Comparing daily 2m Temperature during winter at L2
T87 = num1(:,2);
T_L2 = num1(:,4);
 figure (1)
ax1 = subplot(2,2,2,'Position',[0.55,0.57,0.39,0.39]);
plot(Day,T87,'k',Day,T_L2,'r','LineWidth',2);
title('(c)2m Temperature Comparison at L2, winter','Color','k');
lgd = legend('T87: Temp. 1987','T-L2: CFSR Temperature at L2');
lgd.FontSize = 12;
```

```
xlabel('Time(Day)','FontSize',12);ylabel('Temp (K°)','FontSize',12);
ylim([270 295]);
pbaspect([2.2 1 1]);
% Reading data from excel sheet and plot them for L2
num1 = xlsread('T2_L2_Compare.xlsx','All reading Ave');
Day = num1(:,1);
% Comparing average 2m Temperature during winter at L2
T87 = num1(:,2);
T_L2 = num1(:,4);
 figure (1)
ax1 = subplot(2,2,4,'Position',[0.55,0.13,0.39,0.39]);
plot(Day,T87,'k',Day,T_L2,'r','LineWidth',2);
title('(d)Average 2m Temperature Comparison at L2, winter','Color','k');
lgd = legend('T87: Average Temp.1987','T-L2: CFSR Temp. at L2');
lgd.FontSize = 12;
xlabel('Time(Day)','FontSize',12);ylabel('Temp (K°)','FontSize',12);
ylim([270 295]);
pbaspect([2.2 1 1]);
```

C. Comparing the Wind from WRF Out to CFSR Data

```
clc
clear all
% Reading wind data from excel sheet and plot them for L1
num1 = xlsread('Wind_L1_Compare.xlsx','All Reading');
Day = num1(:,1);
% Comparing wind during summer
AveWind87_1 = num1(:,7);
Wind_L1 = num1(:,4);
figure (1);
ax1W = subplot(1,2,1,'Position',[0.1,0.3,0.39,0.39]);
plot(Day,AveWind87_1,'k',Day,Wind_L1,'r','LineWidth',2);
title('(a)Average Wind Speed comparison at L1','Color','k');
lgd = legend('Wind-87:Average Wind 1987','Wind-L1:CFSR Wind. at L1');
lgd.FontSize = 12;
xlabel('Time (Day)','FontSize',12);ylabel('Wind speed (M/S)','FontSize',12);
ylim([0 12]);
%--------------------------------------------------------------------------
% Reading wind data from excel sheet and plot them for L2
num1 = xlsread('Wind_L2_Compare.xlsx','All Reading');
Day = num1(:,1);
% Comparing wind during summer
AveWind87_2 = num1(:,7);
Wind_L2 = num1(:,4);
figure (1);
ax1 = subplot(1,2,2,'Position',[0.53,0.3,0.39,0.39]);
```

```
plot(Day,AveWind87_2,'k',Day,Wind_L2,'r','LineWidth',2);
title('(b)Average Wind Speed at L2','Color','k');
lgd = legend('Wind-87:Average Wind 1987','Wind-L2:CFSR Wind. at L2');
lgd.FontSize = 12;
xlabel('Time (Day)','FontSize',12);ylabel('Wind speed (M/S)','FontSize',12);
ylim([0 12]);
```

D. Comparing the Relative Humidity from WRF Out to CFSR Data

```
clc
clear all
% Reading Relative Humidity data from excel sheet and plot them for L1
num1 = xlsread('Relative Hu_L1.xlsx','All Readings');
Day = num1(:,1);
% Comparing Relative Humidity during winter
AveRH87_1 = num1(:,7);
RH_L1 = num1(:,4);
figure (1);
ax1W = subplot(1,2,1,'Position',[0.1,0.3,0.39,0.39]);
plot(Day,AveRH87_1,'k',Day,RH_L1,'r','LineWidth',2);
title('(a)Average Relative Humidity comparison at L1','Color','k');
lgd = legend('RH-87:Average Relative Humidity 1987','RH-L1:CFSR Relative Humidity
at L1');
lgd.FontSize = 12;
xlabel('Time (Day)','FontSize',12);ylabel('Relative Humidity (fraction)','FontSize',12);
ylim([0 1.2]);
%--------------------------------------------------------------------------
% Reading Relative Humidity data from excel sheet and plot them for L2
num1 = xlsread('Relative Hu_L2.xlsx','All Readings');
Day = num1(:,1);
% Comparing Relative Humidity during winter
AveRH87_2 = num1(:,7);
RH_L2 = num1(:,4);
figure (1);
ax1 = subplot(1,2,2,'Position',[0.53,0.3,0.39,0.39]);
plot(Day,AveRH87_2,'k',Day,RH_L2,'r','LineWidth',2);
title('(b)Average Relative Humidity comparison at L2','Color','k');
lgd = legend('RH-87:Average Relative Humidity 1987','RH-L2:CFSR Relative Humidity
at L2');
lgd.FontSize = 12;
xlabel('Time (Day)','FontSize',12);ylabel('Relative Humidity (fraction)','FontSize',12);
ylim([0 1.2]);
```

III. Calculating Dust Fluxes

A. Calculating Z_0 , R, and u^*t from Land-Use Maps

- PART 1: CREATION OF 24 CATIGORIES OF LAND SURFACE TYPE MAP

```
clc
clear all
p=60; % x-axiz length
q=60; % y-axis length
% open the file
ncdisp('Geo1987Final.nc');
ncid1 = netcdf.open('Geo1987Final.nc','NC_NOWRITE');
%Latitude and longitude coordinates
long = netcdf.getVar(ncid1,2,[0 0 0],[p q 1]);
lat = netcdf.getVar(ncid1,1,[0 0 0],[p q 1]);
% Land Mask Index
lu = netcdf.getVar(ncid1,24,[0 0 0],[p q 1]);
 latd = double(lat);
longd = double(long);
lu87 = double(lu);
%Plot line dividing land and ocean
hold on
figure (1)
h1=pcolor(longd,latd,lu87);
save('LU_1987.mat','lu87')
% Test values map
size(lu87)
for k = 1:60
   lu87(30,k)
end
```

- PART 2: CREATION OF MAPS LINKS Z_0 , R TO LAND COVER SURFACE MAP

```
clear all;
close all;
z24 = [100 2 2 2 2 2 2 3 3 2 5 5 5 5 5 0.1 0.2 0.3 1 0.3 0.3 0.2 0.1 0.1];
R24 = [1.0 0.4 0.6 0.5 0.5 0.7 0.6 0.7 0.75 0.8 0.9 0.9 0.9 0.9 0.9 1.0 1.0 1.0 0.1 1.0 1.0
1.0 1.0 1.0];
load('LU_1987.mat','lu87');
load('LU_2011.mat','lu11');
z0_1987 = zeros(60,60);
z0_2011 = zeros(60,60);
 for m = 1:60;
   for n = 1:60;
      z0_1987(m,n) = z24(lu87(m,n));
      z0_2011(m,n) = z24(lu11(m,n));
   end
end
```

```
figure (1)
pcolor(z0_1987');
save('Z0_1987.mat','z0_1987');
figure (2)
pcolor(z0_2011');
save('Z0_2011.mat','z0_2011');
Ustart1_1987= zeros(60,60);
Ustart1_2011= zeros(60,60);
for m = 1:60;
   for n = 1:60;
      Ustart1_1987(m,n) = 0.30*exp(7.22*(z0_1987(m,n)));
      Ustart1_2011(m,n) = 0.30*exp(7.22*(z0_2011(m,n)));
   end
end
figure (3)
pcolor(Ustart1_1987');
save('Ustart1_1987.mat','Ustart1_1987');
figure (4)
pcolor(Ustart1_2011');
save('Ustart1_2011.mat','Ustart1_2011');
R0_1987 = zeros(60,60);
R0_2011 = zeros(60,60);
 for m = 1:60;
   for n = 1:60;
      R0_1987(m,n) = R24(lu87(m,n));
      R0_2011(m,n) = R24(lu11(m,n));
   end
end
figure (5)
pcolor(R0_1987');
save('R0_1987.mat','R0_1987');
figure (6)
pcolor(R0_2011');
save('R0_2011.mat','R0_2011');
```

B. Calculating and Creating the Map of Threshold Friction Velocity (*u*t*)

```
clc
clear all
close all
n=240; %% n is time dim length
p=60 % x-axiz length
q=60; % y-axis length
Wprim = ((0.0014*(0.5)^2)+(0.17*0.5)); % Threshold Volumetric Soil Moisture
load('Ustart1_1987.mat','Ustart1_1987');
load('Ustart1_2011.mat','Ustart1_2011');
%% open and read netcdf files for Summer Month for both 1987 and 2011 %%
ncdisp('WRFOutSF_S1987.nc');
ncdisp('WRFOutSF_S2011.nc');
ncid1 = netcdf.open('WRFOutSF_S1987.nc','NC_NOWRITE');
ncid2 = netcdf.open('WRFOutSF_S2011.nc','NC_NOWRITE');
%Latitude and longitude coordinates
long = netcdf.getVar(ncid1,107,[0 0 0],[p q 1]);
lat = netcdf.getVar(ncid1,106,[0 0 0],[p q 1]);
% Reading Moisture from WRF out Summer month
Ws87 = netcdf.getVar(ncid1,52,[0 0 0 0],[p q 1 n]);
Ws11 = netcdf.getVar(ncid2,52,[0 0 0 0],[p q 1 n]);
latd = double(lat);
longd = double(long);
for k = 1:n
   for i = 1:60 ;
   for j = 1:60 ;
      Us87s1(i,j,k) = 0;
       if (Ws87(i,j,k) < Wprim)
          Us87s1(i,j,k) = Ustart1_1987(i,j);
       elseif (Ws87(i,j,k) > Wprim)
          Us87s1(i,j,k) = Ustart1_1987(i,j)*((1+1.21*(Ws87(i,j,k)-Wprim)^0.68)^0.5);
       end
   end
   end
   Us87s1(i,j,k);
   timestamp(k) = k;
end
 save('Us87s1.mat','Us87s1');
for k = 1:n
   for i = 1:60 ;
   for j = 1:60 ;
       Us11s1(i,j,k) = 0;
       if (Ws11(i,j,k) < Wprim)
          Us11s1(i,j,k) = Ustart1_2011(i,j);
          elseif (Ws11(i,j,k) > Wprim)
          Us11s1(i,j,k) = Ustart1_2011(i,j)*((1+1.21*(Ws11(i,j,k)-Wprim)^0.68)^0.5);
```

```
      end
    end
    end
    Us11s1(i,j,k);
    timestamp(k) = k;
end
 save('Us11s1.mat','Us11s1');
```

C. The Final Calculation of Dust Fluxes Based on Maps of Z_0 , R, and Threshold Friction Velocity u^*t

```
clc
clear all
close all
n=240; %% n is time dim length
p=60; % x-axiz length
q=60; % y-axis length
% ######################################################
% Code for accumulated dust fluxes for winter
% ######################################################
%Load required maps for winter:
load('Uw87s1.mat','Uw87s1');% Threshold U* winter 1987
load ('R0_1987.mat','R0_1987'); % R Factor map 1987
load('Uw11s1.mat','Uw11s1');% Threshold U* winter 2011
load ('R0_2011.mat','R0_2011'); % R Factor map 2011
load land1987; % Land use map 1987
load land2011; % Land use map 2011
% First section to draw city border ============
xx = [0 0]; yy = [0 0];
landmask4 = zeros(60,60);
landmask1 = zeros(60,60);
for i = 1:60
   for j = 1:60
      if (lud4(i,j) == 1);
        landmask4(i,j) = 1;
      else;
        landmask4(i,j) = 0;
      end
%
      if (lud1(i,j) == 1);
        landmask1(i,j) = 1;
      else;
        landmask1(i,j) = 0;
      end
   end
end
%
for i = 1:60
   longit(i) = longd4(i,1);
end
for j = 1:60
   latit(j) = latd4(1,j);
end
%===== end of first section ==========
%% open a netcdf file for winter %%
```

```
ncdisp('WRFOutSF_W1987.nc');
ncdisp('WRFOutSF_W2011.nc');
ncid1 = netcdf.open('WRFOutSF_W1987.nc','NC_NOWRITE');
ncid2 = netcdf.open('WRFOutSF_W2011.nc','NC_NOWRITE');
% Reading the Latitude and longitude coordinates:
long = netcdf.getVar(ncid1,107,[0 0 0],[p q 1]);
lat = netcdf.getVar(ncid1,106,[0 0 0],[p q 1]);
% Reading the the Fiction vilocities for winter:
Usta87w = netcdf.getVar(ncid1,118,[0 0 0],[p q n]);
Ustar11w = netcdf.getVar(ncid2,118,[0 0 0],[p q n]);
latd = double(lat);
longd = double(long);
Usta87dw =double(Usta87w*100);
Usta11dw =double(Ustar11w*100);

   for i = 1:60 ;
   for j = 1:60 ;
     Flux87w(i,j) = 0;
     Flux11w(i,j) = 0;
     for k = 1:n;
        if (Usta87dw(i,j,k) > Uw87s1(i,j,k))
          Flux87w(i,j) = Flux87w(i,j)+ (0.13*(1-R0_1987(i,j))*(10^(-
14))*((Usta87dw(i,j,k))^4));
        end

        if (Usta11dw(i,j,k) > Uw11s1(i,j,k))
           Flux11w(i,j) = Flux11w(i,j)+ (0.13*(1-R0_2011(i,j))*(10^(-
14))*((Usta11dw(i,j,k))^4));
        end
     end
  end
  end
% Accumulated dust flux 1987 plot
figure (1)
ax1 = subplot(2,2,1,'Position',[0.18,0.57,0.39,0.39]);
hold on
contourf(longd,latd,Flux87w);
% Second section to draw city border 1987 ========
% ----- draw vertical segments -----
for j = 1:60
   for i = 1:60-1
      if (abs(landmask4(i,j)-landmask4(i+1,j)) > 0.1);
         xx(1) = (longit(i+1)+longit(i))/2;
         xx(2) = xx(1);
         if (j == 60);
           yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
```

```
            yy(2) = latit(j);
          elseif (j == 1);
            yy(1) = latit(1);
            yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
          else;
            yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
            yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
          end
          plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
       end
    end
end
% ----- draw horizontal segments -----
for i = 1:60
    for j = 1:60-1
       if (abs(landmask4(i,j)-landmask4(i,j+1)) > 0.1);
          yy(1) = (latit(j+1)+latit(j))/2;
          yy(2) = yy(1);
          if (i == 60);
            xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
            xx(2) = longit(i);
          elseif (i == 1);
            xx(1) = longit(1);
            xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
          else
            xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
            xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
          end
          plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
       end
    end
end
% -------------------------
hold off
% ==== end of second section for border 1987========
title({'(a)1987 Winter'},'Color','k');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
colorbar('eastoutside','FontSize',12);
c = colorbar;
caxis([0 3*10^(-6)]);
c.Label.String = 'Dust Flux Intensity (g/cm^2.s)';
c.FontSize = 12;
pbaspect([1 1 1])
% Accumulated dust flux plot for 2011
figure (1);
ax1 = subplot(2,2,3,'Position',[0.18,0.07,0.39,0.39]);
```

```
hold on
contourf(longd,latd,Flux11w);
% Second section to draw city border 2011 ========
% ----- draw vertical segments -----
for j = 1:60
   for i = 1:60-1
      if (abs(landmask1(i,j)-landmask1(i+1,j)) > 0.1);
         xx(1) = (longit(i+1)+longit(i))/2;
         xx(2) = xx(1);
         if (j == 60);
           yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
           yy(2) = latit(j);
         elseif (j == 1);
           yy(1) = latit(1);
           yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
         else;
           yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
           yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
         end
         plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
      end
   end
end
% ----- draw horizontal segments -----
for i = 1:60
   for j = 1:60-1
      if (abs(landmask1(i,j)-landmask1(i,j+1)) > 0.1);
         yy(1) = (latit(j+1)+latit(j))/2;
         yy(2) = yy(1);
         if (i == 60);
           xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
           xx(2) = longit(i);
         elseif (i == 1);
           xx(1) = longit(1);
           xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
         else
           xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
           xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
         end
         plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
      end
   end
end
% -------------------------
hold off
% ==== end of second section to draw borders for 2011========
```

```
title({'(b)2011 Winter'},'Color','k');
xlabel('Longitude(°)');ylabel('Latitude(°)');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
colorbar('eastoutside','FontSize',12);
c = colorbar;
caxis([0 3*10^(-6)]);
c.Label.String = 'Dust Flux Intensity (g/cm^2.s)';
c.FontSize = 12;
pbaspect([1 1 1]);
% ####################################################
% Code for accumulated dust fluxes for summer
% ####################################################
%Load required maps for summer:
load('Us87s1.mat','Us87s1');% Threshold U* summer 1987
load ('R0_1987.mat','R0_1987'); % R Factor map 1987
load('Us11s1.mat','Us11s1');% threshold U* summer 2011
load ('R0_2011.mat','R0_2011'); % R Factor map 2011
% First section to draw city border ============
load land1987; % Land use map 1987
load land2011; % Land use map 2011
xx = [0 0]; yy = [0 0];
landmask4 = zeros(60,60);
landmask1 = zeros(60,60);
for i = 1:60
   for j = 1:60
      if (lud4(i,j) == 1);
        landmask4(i,j) = 1;
      else;
        landmask4(i,j) = 0;
      end
%
      if (lud1(i,j) == 1);
        landmask1(i,j) = 1;
      else;
        landmask1(i,j) = 0;
      end
   end
end
%
for i = 1:60
   longit(i) = longd4(i,1);
end
for j = 1:60
   latit(j) = latd4(1,j);
end
%===== end of first section  ==========
```

```
% Open a netcdf file for summer%%
ncdisp('WRFOutSF_S1987.nc');
ncdisp('WRFOutSF_S2011.nc');
ncid1 = netcdf.open('WRFOutSF_S1987.nc','NC_NOWRITE');
ncid2 = netcdf.open('WRFOutSF_S2011.nc','NC_NOWRITE');
%Reading the latitude and longitude coordinates:
long = netcdf.getVar(ncid1,107,[0 0 0],[p q 1]);
lat = netcdf.getVar(ncid1,106,[0 0 0],[p q 1]);
%Reading the friction velocity:
Usta87w = netcdf.getVar(ncid1,118,[0 0 0],[p q n]);
Ustar11w = netcdf.getVar(ncid2,118,[0 0 0],[p q n]);
latd = double(lat);
longd = double(long);
Usta87dw =double(Usta87w*100);
Usta11dw =double(Ustar11w*100);
   for i = 1:60 ;
   for j = 1:60 ;
     Flux87w(i,j) = 0;
     Flux11w(i,j) = 0;
     for k = 1:n;
        if (Usta87dw(i,j,k) > Us87s1(i,j,k))
          Flux87w(i,j) = Flux87w(i,j)+ (0.13*(1-R0_1987(i,j))*(10^(-
14))*((Usta87dw(i,j,k))^4));
        end

        if (Usta11dw(i,j,k) > Us11s1(i,j,k))
           Flux11w(i,j) = Flux11w(i,j)+ (0.13*(1-R0_2011(i,j))*(10^(-
14))*((Usta11dw(i,j,k))^4));
        end
     end
  end
  end
% Accumulated summer dust flux 1987 plot
figure (1)
ax1 = subplot(2,2,2,'Position',[0.51,0.57,0.39,0.39]);
hold on
contourf(longd,latd,Flux87w);
% Second section to draw city border 1987 ========
% ----- draw vertical segments -----
for j = 1:60
   for i = 1:60-1
      if (abs(landmask4(i,j)-landmask4(i+1,j)) > 0.1);
         xx(1) = (longit(i+1)+longit(i))/2;
         xx(2) = xx(1);
         if (j == 60);
           yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
```

```
            yy(2) = latit(j);
          elseif (j == 1);
            yy(1) = latit(1);
            yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
          else
            yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
            yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
          end
          plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
       end
    end
end
% ----- draw horizontal segments -----
for i = 1:60
    for j = 1:60-1
       if (abs(landmask4(i,j)-landmask4(i,j+1)) > 0.1);
          yy(1) = (latit(j+1)+latit(j))/2;
          yy(2) = yy(1);
          if (i == 60);
            xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
            xx(2) = longit(i);
          elseif (i == 1);
            xx(1) = longit(1);
            xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
          else
            xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
            xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
          end
          plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
       end
    end
end
% -------------------------
hold off
% ==== end of second section for border drawing 1987 ========
title({'(c)1987 Summer'},'Color','k');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
colorbar('eastoutside','FontSize',12);
c = colorbar;
caxis([0 3*10^(-6)]);
c.Label.String = 'Dust Flux Intensity (g/cm^2.s)';
c.FontSize = 12;
pbaspect([1 1 1])
% Accumulated summer dust flux 2011 plot:
figure (1);
ax1 = subplot(2,2,4,'Position',[0.51,0.07,0.39,0.39]);
```

```
hold on
contourf(longd,latd,Flux11w);
% Second section to draw city border 2011 ========
% ----- draw vertical segments -----
for j = 1:60
   for i = 1:60-1
      if (abs(landmask1(i,j)-landmask1(i+1,j)) > 0.1);
         xx(1) = (longit(i+1)+longit(i))/2;
         xx(2) = xx(1);
         if (j == 60);
           yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
           yy(2) = latit(j);
         elseif (j == 1);
           yy(1) = latit(1);
           yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
         else;
           yy(1) = latit(j)-(latit(j)-latit(j-1))/2;
           yy(2) = latit(j)+(latit(j+1)-latit(j))/2;
         end
         plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
      end
   end
end
% ----- draw horizontal segments -----
for i = 1:60
   for j = 1:60-1
      if (abs(landmask1(i,j)-landmask1(i,j+1)) > 0.1);
         yy(1) = (latit(j+1)+latit(j))/2;
         yy(2) = yy(1);
         if (i == 60);
           xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
           xx(2) = longit(i);
         elseif (i == 1);
           xx(1) = longit(1);
           xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
         else
           xx(1) = longit(i)-(longit(i)-longit(i-1))/2;
           xx(2) = longit(i)+(longit(i+1)-longit(i))/2;
         end
         plot(xx,yy,'Color',[0.9 0.9 0.6],'LineWidth',2)
      end
   end
end
% -------------------------
hold off
% ==== end of second section fro border drawing 2011========
```

```
title({'(d)2011 Summer'},'Color','k');
xlabel('Longitude(°)');ylabel('Latitude(°)');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
colorbar('eastoutside','FontSize',12);
c = colorbar;
caxis([0 3*10^(-6)]);
c.Label.String = 'Dust Flux Intensity (g/cm^2.s)';
c.FontSize = 12;
pbaspect([1 1 1]);
```

D. Calculation of In-Depth Analysis for (u^*), Soil Moisture (w), and Dust Flux (D_f) of Specific Location Erbil 1987 Case

```
clear all;
close all;
clc
n=240; %% n is time dim length
p=60 % x-axiz length
q=60; % y-axis length
s=4; % Soil Moisture Layers
load('Us87s1.mat','Us87s1');% Threshold U*t Summer 1987
load ('R0_1987.mat','R0_1987'); % R Factor map 1987
load('Us11s1.mat','Us11s1');% Threshold U*t Summer 2011
load ('R0_2011.mat','R0_2011'); % R Factor map 2011
%% open a netcdf file %%
ncdisp('WRFOutSF_S1987.nc');
ncdisp('WRFOutSF_S2011.nc');
ncid1 = netcdf.open('WRFOutSF_S1987.nc','NC_NOWRITE');
ncid2 = netcdf.open('WRFOutSF_S2011.nc','NC_NOWRITE');
U87 = netcdf.getVar(ncid1,118,[0 0 0],[p q n]);
U11 = netcdf.getVar(ncid2,118,[0 0 0],[p q n]);
ustd87 = double(U87*100);
ustd11 = double(U11*100);
  for i = 37;
   for j = 27 ;
     for k = 1:n;
       ustmax87(k) = ustd87(i,j,k);
       Ucritical87(k)= Us87s1(i,j,k);
       if (ustd87(i,j,k) > Us87s1(i,j,k))
          Flux87s(k) =  (0.13*(1-R0_1987(i,j))*(10^(-14))*((ustd87(i,j,k))^4));
          hold on
        elseif (ustd87(i,j,k) < Us87s1(i,j,k))
            Flux87s(k) = 0;
       end
       ustmax11(k) = ustd11(i,j,k);
       Ucritical11(k)= Us11s1(i,j,k);
       if (ustd11(i,j,k) > Us11s1(i,j,k))
          Flux11s(k) =  (0.13*(1-R0_2011(i,j))*(10^(-14))*((ustd11(i,j,k))^4));
          hold on;
        elseif (ustd11(i,j,k) < Us11s1(i,j,k))
            Flux11s(k) = 0;
       end
   ustmax87(k);
   Ucritical87(k);
   Flux87s(k) ;
   ustmax11(k);
   Ucritical11(k);
```

```
    Flux11s(k) ;
    timestamp(k) = k;
          end
      end
     end
figure (1)
plot(timestamp,ustmax87,'k',timestamp,Ucritical87,'r','LineWidth',2);
title('Friction Velocity during Summer July, 2000 City size on 1987','Color','k');
lgd = legend('Ustar87','Ucritical87');
lgd.FontSize = 10;
xlabel('Time Step:U* reading every three hours for 30 days, ');ylabel('Friction Velocity
(U*)(cm/s)');
figure (2)
plot(timestamp,ustmax11,'k',timestamp,Ucritical11,'r','LineWidth',2);
title('Friction Velocity profile during Summer July, 2000 City size on 2011','Color','k');
lgd = legend('Ustar11','Ucritical11');
lgd.FontSize = 10;
xlabel('Time Step:U* reading every three hours for 30 days, ');ylabel('Friction Velocity
(U*)(cm/s)');
Ws87d = netcdf.getVar(ncid1,52,[0 0 0 0],[p q s n]);
Ws11d = netcdf.getVar(ncid2,52,[0 0 0 0],[p q s n]);
Ws87 =double(Ws87d);
Ws11 =double(Ws87d);

for i = 30;
     for j = 40 ;
     for v = 1
       for k = 1:n;
         Wstmax87(k) = Ws87(i,j,v,k);
         Wstmax11(k) = Ws11(i,j,v,k);
    Wstmax87(k);
    Wstmax11(k);
   timestamp(k) = k;
       end
      end
    end
end
figure (3)
plot(timestamp,Wstmax87,'r','LineWidth',2);
title('Soil Moisture profile during Summer July, 2000 City size on 1987','Color','k');
xlabel('Time Step:W reading every three hours for 30 days, ');ylabel('Soil Moisture
(W)(m^3/m^3)');
figure (4)
plot(timestamp,Wstmax11,'r','LineWidth',2);
title('Soil Moisture profile during Summer July, 2000 City size on 2011','Color','k');
xlabel('Time Step:W reading every three hours for 30 days, ');ylabel('Soil Moisture
```

```
((W)(m^3/m^3)');
figure (5)
plot(timestamp,Flux87s,'r','LineWidth',2)
title('Dust flux profile for the month of July 2000 for Erbil city size 1987','Color','k')
xlabel('Time Step:Dust flux reading every three hours for 30 days, ');ylabel('Dust Flux
Intensity (g/cm^2.s)');
figure (6)
plot(timestamp,Flux11s,'r','LineWidth',2)
title('Dust flux profile for the month of July 2000 for Erbil city size 2011','Color','k')
xlabel('Time Step:Dust flux reading every three hours for 30 days, ');ylabel('Dust Flux
Intensity (g/cm^2.s)');
```

IV. Plots of Dust Flux (D_f) Transport for Specific Locations Erbil 1987 Case

```
clc
clear all
p=60; % x-axiz length
q=60; % y-axis length
% Plotting the dust particles' path
% open the file
ncdisp('WRF 1987_WinShort.nc');
ncid87 = netcdf.open('WRF 1987_WinShort.nc','NC_NOWRITE');
% Reading Latitude and longitude coordinates 1987
long87 = netcdf.getVar(ncid87,107,[0 0 0],[p q 1]);
lat87 = netcdf.getVar(ncid87,106,[0 0 0],[p q 1]);
% Reading Land Mask Index 1987
lu87 = netcdf.getVar(ncid87,1,[0 0 0],[p q 1]);
latd87 = double(lat87);
longd87 = double(long87);
lud87 = double(lu87);
% Plot the maps and the dust transport 1987
figure (1);
ax2 = subplot(1,2,1,'Position',[0.01,0.27,0.56,0.56]);
x_87 =[44.16 43.93 43.805];      y_87 =[36.24 36.40 36.46];
x1_87 =[44.16 43.91  43.866];    y1_87 =[36.24 36.43 36.46];
x2_87 =[44.16 43.93 43.934];     y2_87 =[36.24 36.45 36.46];
x3_87 =[44.16 43.93];            y3_87 =[36.24 36.46];
x4_87 =[44.08 43.846];           y8_87 =[36.1 36.28];
x5_87 =[44.08 43.84];            y9_87 =[36.1 36.33];
x6_87 =[44.08 43.85];            y10_87 =[36.1 36.43];
x7_87 =[44.08 43.96];            y11_87 =[36.1 36.46];
h87=pcolor(longd87,latd87,lud87);
set(h87,'EdgeColor','none');
title('(a)LULC, 1987','Color','k');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12);
caxis([1 9]);

hold on
line(x_87,y_87,'LineStyle',':','Color',[0.9,0.5,0.5],'LineWidth',2);
line(x1_87,y1_87,'LineStyle','-.','Color',[0,0,0.1],'LineWidth',2);
line(x2_87,y2_87,'LineStyle',':','Color',[1,0,0],'LineWidth',2);
line(x3_87,y3_87,'LineStyle','-.','Color',[0,1,0],'LineWidth',2);
line(x4_87,y8_87,'Color',[0.9,0.5,0.5],'LineWidth',2);
line(x5_87,y9_87,'Color',[0,0,0.1],'LineWidth',2);
line(x6_87,y10_87,'Color',[1,0,0],'LineWidth',2);
line(x7_87,y11_87,'Color',[0,1,0],'LineWidth',2);
hold on
s87 = [43.843 43.84 43.85] ;
```

```
t87 = [36.28 36.33 36.43];
plot(s87,t87,'*r');
labels = {'P1','P2','P3'};
text(s87,t87,labels,'color','k','VerticalAlignment','top','HorizontalAlignment','right','FontSi
ze',12);
hold on
hold on
x87 = [44.16 44.08] ;
y87 = [36.24 36.1];
labels = {'LT1','LT2'};
plot (x87,y87,'*w');
text(x87,y87,labels,'color','k','VerticalAlignment','top','HorizontalAlignment','left','FontSi
ze',8);
pbaspect([4 4 4]);
%----------------------------------------
% Plotting the dust particles' path 2011
% open the file 2011
ncdisp('WRF 2011_WinShort.nc');
ncid11 = netcdf.open('WRF 2011_WinShort.nc','NC_NOWRITE');
% Reading the Latitude and longitude coordinates 2011
long11 = netcdf.getVar(ncid11,107,[0 0 0],[p q 1]);
lat11 = netcdf.getVar(ncid11,106,[0 0 0],[p q 1]);
% Reading the Land Mask Index 2011
lu11 = netcdf.getVar(ncid11,1,[0 0 0],[p q 1]);
latd11 = double(lat11);
longd11 = double(long11);
lud11 = double(lu11);
% Plot the maps and the dust transport 2011
figure (1);
ax2 = subplot(1,2,2,'Position',[0.38,0.27,0.56,0.56]);
x_11 =[44.16 43.96 43.822];     y_11 =[36.24 36.39 36.46];
x1_11 =[44.16 43.94 43.912];   y1_11 =[36.24 36.43 36.46];
x2_11 =[44.16 43.93 43.93];     y2_11 =[36.24 36.45 36.46];
x3_11 =[44.16 43.94];              y3_11 =[36.24 36.45];
x4_11 =[44.08 43.84];              y8_11 =[36.1 36.29];
x5_11 =[44.08 43.84];              y9_11 =[36.1 36.36];
x6_11 =[44.08 43.88];             y10_11 =[36.1 36.42];
x7_11 =[44.08 43.98];             y11_11 =[36.1 36.46];

h1=pcolor(longd11,latd11,lud11);
colorbar('Ticks',[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24],...
   'TickLabels',{'1', '2','3','4','5','6','7','8','9'},'FontSize',12);
set(h1,'EdgeColor','none');
title('(b)LULC, 2011','Color','k');
xlabel('Longitude(°)','FontSize',12);ylabel('Latitude(°)','FontSize',12)
caxis([1 9]);
```

```
hold on
line(x_11,y_11,'LineStyle','-.','Color',[0.9,0.5,0.5],'LineWidth',2);
line(x1_11,y1_11,'LineStyle','-.','Color',[0,0,0.1],'LineWidth',2);
line(x2_11,y2_11,'LineStyle',':','Color',[1,0,0],'LineWidth',2);
line(x3_11,y3_11,'LineStyle','-.','Color',[0,1,0],'LineWidth',2);
line(x4_11,y8_11,'Color',[0.9,0.5,0.5],'LineWidth',2);
line(x5_11,y9_11,'Color',[0,0,0.1],'LineWidth',2);
line(x6_11,y10_11,'Color',[1,0,0],'LineWidth',2);
line(x7_11,y11_11,'Color',[0,1,0],'LineWidth',2);
hold on
s11 = [43.84 43.84 43.88] ;
t11 = [36.29 36.36 36.42];
plot(s11,t11,'*r');
labels = {'P1','P2','P3'};
text(s11,t11,labels,'color','k','VerticalAlignment','top','HorizontalAlignment','right','FontSi
ze',12);
hold on
x11 = [44.16 44.08] ;
y11 = [36.24 36.1];
labels = {'LT1','LT2'};
plot (x11,y11,'*w');
text(x11,y11,labels,'color','k','VerticalAlignment','top','HorizontalAlignment','left','FontSi
ze',8);
pbaspect([4 4 4]);
```

BIOGRAPHICAL SKETCH

Sherzad Tahseen Tahir was borne in Erbil-Iraq then raised in Baghdad. He completed his undergraduate degree in Mechanical Engineering at the University of Mosul in Iraq. Fleeing Iraq to Azerbaijan, he spent time there working with various humanitarian NGO’s. He moved to the United State of America by a resettlement asylum program and worked for Micron Technology Inc., Boise-Idaho for almost eight years. Then, he earned his Master of Engineering degree in Mechanical Engineering from Boise State University, Boise-Idaho. Following this, he moved to Iraq and worked as teacher assistant at the College of Engineering, University of Duhok, Duhok-Iraq, In the latest part of this journey, Sherzad joined the School for Engineering of Matter, Transport, and Energy, Arizona State University, Tempe-Arizona. Sherzad's research interests are in the fields of fluid dynamics and the environment.