

# MOVES-Matrix Running Module Quick Start Guide

Randall Guensler  
Haobing Liu  
Hongyu Lu

Georgia Institute of Technology  
School of Civil and Environmental Engineering

# Quick Start Guide

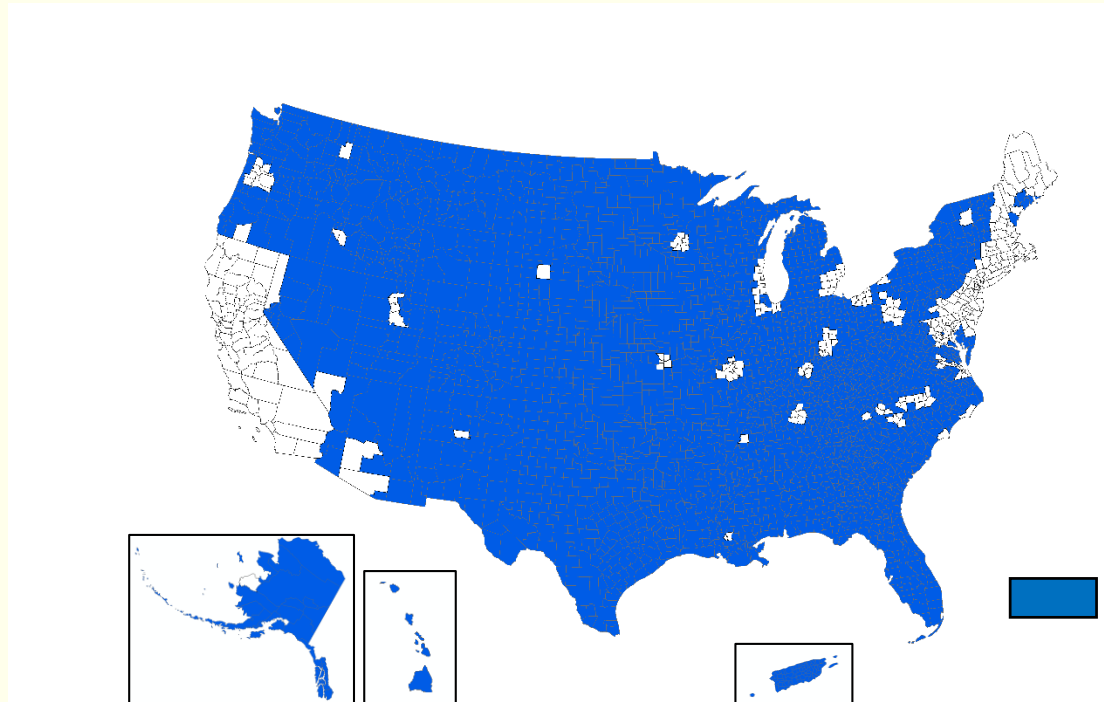
- Slides provide a quick start guide for using the MOVES-Matrix running module to analyze project-level emissions
  - For experienced MOVES modelers
- Emission rates are derived from MOVES2014b project-level runs iterated across input variables
- The accompanying data files specifically reflect:
  - The five regions' MOVES default fuel supply
  - The five regions' MOVES default I/M program

# MOVES-Matrix Background

- **MOVES-Matrix provides fuel consumption and running exhaust emissions for project-level analyses and county-level runs**
- **The MOVES-Matrix database is populated with the results from more than 100,000 iterative MOVES runs**
- **MOVES-Matrix yields the exact same values obtained by running MOVES, but runs in a fraction of the time**
- **Users apply specific emission rates to each link in the transportation network via a scripting process**

# MOVES-Matrix Current Coverage

(Updated Sep 2021)



# Python is Required for MOVES-Matrix

- **Install Python 2.7 on your computer**
- **Option 1 (recommended)**
  - **Install Anaconda2 (for python2.7) (which includes needed modules)**
  - **<https://www.continuum.io/downloads>**
- **Option 2**
  - **Install Python**
  - **Add required modules (csv, os, math, time, sys, and pandas)**

# Six File Sets are Provided in the “MOVES-Matrix” Folder

- **MOVES-matrix Python program**
  - moves\_matrix\_py27\_113018.py
- **Batch mode allocation spreadsheet**
  - batchmode.csv
- **MOVES-Matrix emission rate database**
  - MatrixData
- **Input CSV files in the ‘input’ folder**
  - Standard set of MOVES input files
- **Output CSV files in the ‘output’ folder**
  - Emission inventory and emission rates
- **OpMode Distribution from MOVES default cycles**
  - default\_opmode\_project.csv

# Change the Python Directory Paths

- Edit the Python code
  - Use Spyder if you installed Anaconda, or
  - Use any standard text editor (e.g., notepad)
- In the first lines, set the path for your MOVES-Matrix working directory and your database directory:

```
## this path may need to be changed to locate matrix working directory  
path = "E:\\MMatrix_Running Module"  
## this path may need to be changed to locate matrix database directory  
matrixdatapath = "E:\\MMatrix_Running Module\\MatrixData"
```

# Running MOVES-Matrix

**Step 1: Prepare input files (can be scripted)**

**Step 2: Set up batch mode tasks**

**Step 3: Start the Python program**



# Step 1: Input File Preparation

- **Prepare input files in 'input' folder as if running MOVES at the project level**
  - **csv files for link information**  
**Same as the MOVES link input module**
  - **csv files for link source type distribution**  
**Same as MOVES linksourcetypehour input module**
  - **csv files for age distribution**  
**Same as the MOVES sourcetypeagedistribution input module**

# Step 1: Input File Preparation

- **Prepare input files in the 'input' folder as if running MOVES at project level**
  - **csv files for meteorology information**  
**Same as the MOVES meteorology input module**
    - **Set temperature between 10-110°F in 5°F bins (10,15,20...)**
    - **Set humidity between 0-100% in 5% bins (0,5,10...)**

# Step 1: Input File Preparation

- Prepare input files in the 'input' folder as if running MOVES at the project level
  - csv files for operating information
- If you use customized operating input data
  - OpMode distributions, similar to the opmodedistribution module in MOVES (see slide 16 for details)
  - Or, drive schedules, same as driveschedulesecondlink input module in MOVES
- You don't need to prepare operating input, if you want to obtain emissions based on MOVES default driving cycles (default\_opmode\_project.csv is applied here)

# Step 1: Input File Preparation

- Modeled transportation links (link module)
- Source type distributions (linksourcetypehour module)
- Fleet age (sourcetypeagedistribution module)
- Meteorology (meteorology module)
- Operating mode distributions
  - If you use customized operating inputs
    - opmodedistribution module
    - driveschedulesecondlink module

# Prepare the Link File

- **Modeled transportation links**

linkID	countyID	zoneID	roadTypeID	linkLength	linkVolume	linkAvgSpeed	linkDescription	linkAvgGrade
1	13121	131210	4	10	1200	55	Urban restricted	0
2	13121	131210	5	5	600	20	Urban unrestricted	0

- **linkID: link ID**
- **countyID: county ID (not used in MOVES-Matrix)**
- **zoneID: zone ID (not used in MOVES-Matrix)**
- **roadTypeID: road type ID (needed for MOVES default cycle)**
- **linkLength: link length in mile**
- **linkVolume: link volume in # of vehicles**
- **linkAvgSpeed: link average speed in mph  
Not needed if driving schedule is provided)**
- **linkDescription: link description (not used)**
- **linkAvgGrade: link road grade (not used)**

# Prepare Source Type and Age Distributions

- **Source type distributions**

- linkID: link ID
- sourceTypeID: source use type ID
- sourceTypeHourFraction: operating time fraction (usually sum to 1 within each link)

linkID	sourceTypeID	sourceTypeHourFraction
1	11	0.028857869
1	21	0.453908367
1	31	0.311555405

- **Age distributions**

- sourceTypeID: source use type ID
- yearID: calendar year
- ageID: vehicle age in years
- ageFraction: fleet age fraction (usually sum to 1 within each source type)

sourceTypeID	yearID	ageID	ageFraction
11	2014	0	0.071822023
11	2014	1	0.068843178
11	2014	2	0.06477528
11	2014	3	0.051962457
11	2014	4	0.046925796
11	2014	5	0.048601487
11	2014	6	0.086120656
11	2014	7	0.076624358
11	2014	8	0.072596604

# Prepare Meteorology Inputs

## ● Meteorology

- **monthID:** month (1-12)
- **zoneID:** zone ID (not used)
- **hourID:** hour (0-23, not used)
- **temperature:** in F
  - Set temperature between 0-110°F in 5°F bins (0,5,10...)
- **relHumidity:** in %
  - Set humidity between 0-100% in 5% bins (0,5,10...)

monthID	zoneID	hourID	temperature	relHumidity
1	131210	9	45	60

# Prepare Operating Mode Inputs

## OpMode Distribution or Drive Schedule

- **opmodedistribution**

- **sourceTypeID**: source use type ID
- **linkID**: link ID
- **opModeID**: operating mode bin
- **opModeFraction**: fraction of operating mode bin (usually sum to 1 within each source type and link)

sourceTypeID	linkID	opModeID	opModeFraction
11	1	0	0.04932
11	1	1	0.013572
11	1	11	0.019541
11	1	12	0.01788
11	1	13	0.007915

- **driveschedulesecondlink**

- **linkID**: link ID
- **secondID**: time series (seconds from start)
- **speed**: second-by-second speed (mph)
- **grade**: second-by-second road grade (%)

linkID	secondID	speed	grade
1	0	46.7	0
1	1	46.9	0
1	2	47.2	0
1	3	47.8	0
1	4	48.2	0
1	5	49	0



# Step 2: Set Up Batch Mode Tasks

- Open `batchmode.csv` in the 'MOVES-Matrix' directory
- Fill in corresponding input file names within each row
  - Each row represents one task
- Users may enter input file name with or without the '.csv' extension

# Preparing Batch Mode Tasks File

taskID	region	calendarYear	meteorologyFileName	sourceTypeDistributionFileName	ageDistributionFileName	linkFileName	driveSchedule/OpModeDistribution	opmode(o)/cycle(d)/speed(v)
0	atlanta	2014	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2014.csv	ageDistribution_2014.csv	link_ATL_13121.csv		v
1	atlanta	2014	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2014.csv	ageDistribution_2014.csv	link_ATL_13121.csv	drivingCycle.csv	d
2	atlanta	2014	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2014.csv	ageDistribution_2014.csv	link_ATL_13121.csv	opModeDistribution_MOVESMatrix	o
3	atlanta	2018	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2018.csv	ageDistribution_2018.csv	link_ATL_13121.csv		v
4	atlanta	2018	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2018.csv	ageDistribution_2018.csv	link_ATL_13121.csv	drivingCycle.csv	d
5	atlanta	2018	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2018.csv	ageDistribution_2018.csv	link_ATL_13121.csv	opModeDistribution_MOVESMatrix	o

- **taskID:** task ID, similar to MOVESRunID in MOVES
- **region:** enter one of the five regions:
  - denver, atlanta, buffalo, dc, seattle
- **calendarYear:** calendar year
- **meteorologyFileName:** meteorology csv input file
- **sourceTypeDistributionFileName:** source type distribution input file

# Preparing the Batch Mode Tasks File


taskID	region	calendarYear	meteorologyFileName	sourceTypeDistributionFileName	ageDistributionFileName	linkFileName	driveSchedule/OpModeDistribution	opmode(o)/cycle(d)/speed(v)
0	atlanta	2014	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2014.csv	ageDistribution_2014.csv	link_ATL_13121.csv		v
1	atlanta	2014	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2014.csv	ageDistribution_2014.csv	link_ATL_13121.csv	drivingCycle.csv	d
2	atlanta	2014	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2014.csv	ageDistribution_2014.csv	link_ATL_13121.csv	opModeDistribution_MOVESMatrix	o
3	atlanta	2018	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2018.csv	ageDistribution_2018.csv	link_ATL_13121.csv		v
4	atlanta	2018	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2018.csv	ageDistribution_2018.csv	link_ATL_13121.csv	drivingCycle.csv	d
5	atlanta	2018	meteorology_ATL_13121_07.csv	sourceTypeDistribution_2018.csv	ageDistribution_2018.csv	link_ATL_13121.csv	opModeDistribution_MOVESMatrix	o

- **ageDistributionFileName:** age distribution input file
- **linkFileName:** link input file
- **driveSchedule/opModeDistributionFileName:** drive schedule file or opMode distribution file
- **opmode(o)/cycle(d)/speed(v):**
  - ‘d’ if drive schedule is used
  - ‘o’ if opmode distribution is used
  - ‘v’ to apply MOVES default cycles

# opmode(o)/cycle(d)/speed(v) Method

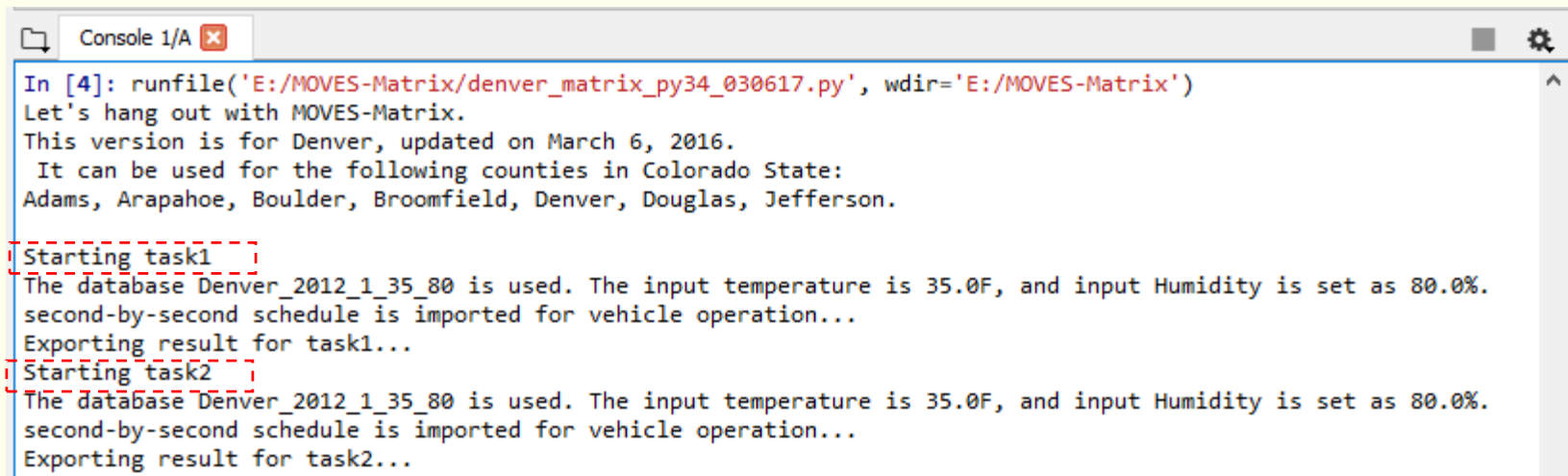
- If 'd' is chosen: drive schedule input is needed
- If 'o' is chosen: opmode distribution input is needed. Average speed in link input file is needed
- If 'v' is chosen: average speed and road type ID is needed in link input file (no operating inputs needed)
  - Average speed: choose from 1-80 mph in 0.1 mph interval
  - Road type ID: 2-Rural restricted access, 3-Rural unrestricted access, 4-Urban restricted access, 5-Urban unrestricted access

# Step 3: Start the Python Program

- Run the program `moves_matrix_py27_113018.py`
- Option 1: Use Anaconda (recommended):  
Use Spyder to open python code, click  to run
- Option 2: Run python code through command line  
Change directory to MOVES-Matrix folder path,  
type `'python moves_matrix_py27_113018.py'`  
and press ENTER

# Calculation Process

- Program takes several seconds to finish calculations
- Status of each task is shown on the program screen



```
Console 1/A x
In [4]: runfile('E:/MOVES-Matrix/denver_matrix_py34_030617.py', wdir='E:/MOVES-Matrix')
Let's hang out with MOVES-Matrix.
This version is for Denver, updated on March 6, 2016.
It can be used for the following counties in Colorado State:
Adams, Arapahoe, Boulder, Broomfield, Denver, Douglas, Jefferson.
Starting task1
The database Denver_2012_1_35_80 is used. The input temperature is 35.0F, and input Humidity is set as 80.0%.
second-by-second schedule is imported for vehicle operation...
Exporting result for task1...
Starting task2
The database Denver_2012_1_35_80 is used. The input temperature is 35.0F, and input Humidity is set as 80.0%.
second-by-second schedule is imported for vehicle operation...
Exporting result for task2...
```

# Visually Verify Input File Information

- For each task, MOVES-Matrix grabs a specific emission rate table from the matrix database, based upon region, calendar year (in batchmode.csv), month, and meteorology (in meteorology input file)
- The database table that will be used in the each task analysis is displayed on the program screen.

```
Starting task1
The database Denver_2012_1_35_80 is used. The input temperature is 35.0F, and input Humidity is set as 80.0%.
second-by-second schedule is imported for vehicle operation...
Exporting result for task1...
Starting task2
The database Denver_2012_1_35_80 is used. The input temperature is 35.0F, and input Humidity is set as 80.0%.
second-by-second schedule is imported for vehicle operation...
Exporting result for task2...
```

# Files Generated

- The output file in the working directory will contain four csv files generated by the Python script
  - taskX\_emissionbylink.csv
  - taskX\_emissionbylinksource.csv
  - X is the task number defined in batchmode.csv



# Output Tables

- **taskX\_emissionbylink.csv**
  - **Column[emquant]:** Emissions and energy consumption (g or KJ) for each link in each task (similar to MOVES results in the ‘movesoutput’ table)
  - **Column[emrate]:** Emission and energy consumption rates (g/mile or KJ/mile) for each link in each task (similar to MOVES results in the ‘rateperdistance’ table)

# Output Tables

- **taskX\_emissionbylinksource.csv**
  - **Column[emquant]: Emissions and energy consumption (g or KJ) for each source type within each link in each task (similar to MOVES results in the ‘movesoutput’ table when disaggregated by source type)**
  - **Column[emrate]: Emission and energy consumption rates (g/mile/vehicle or KJ/mile/vehicle) for each source type within each link in each task**

# Scripting Input Files to Handle Network Runs

- **Users can create input files using Python or Perl for simulation model runs or complex corridor analysis**
  - **Example: Atlanta Travel Demand Model: 74,000+ links**  
**Obtain volume and speed for each link and assign corresponding operation and fleet composition**
- **When drive schedule mode is used, all vehicles on the link are assigned the drive schedule provided**
- **When opmode distribution mode is used, each source type can be assigned a different opmode distribution on each link**
- **When MOVES default cycle mode is used, all vehicles on a link are assigned the same average speed**

# MOVES-Matrix Quick Start Summary

- **The Georgia Tech research team has provided:**
  - **MOVES-Matrix database for five regions**
  - **MOVES-Matrix Python scripts**
- **MOVES users can easily implement MOVES-Matrix**
  - **Very fast, high-resolution modeling runs**
  - **Users can develop scripts to create input files**
  - **Users can script multiple runs**

# References

- **Guensler, R., H. Liu, X. Xu, Y. Xu, and M.O. Rodgers (2016). MOVES-Matrix: Setup, Implementation, and Application. 95th Annual Meeting of the Transportation Research Board. Washington, DC. January 2016**
- **Xu, X., H. Liu, Y. Xu, M. Hunter, M.O. Rodgers and R. Guensler (2016). Estimating Project-Level Vehicle Emissions with VISSIM and MOVES-Matrix. Transportation Research Record, No. 2570, pp. 107-117.**
- **Liu, H., R. Guensler, H. Lu, Y. Xu, X. Xu, and M.O. Rodgers (2019). MOVES-Matrix for High-Performance On-Road Energy and Running Emission Rate Modeling Applications. Journal of the Air & Waste Management Association, July 2019, 10962247.2019.1640806. <https://doi.org/10.1080/10962247.2019.1640806>.**
- **Liu, H., X. Xu, M.O. Rodgers, Y. Xu, R. Guensler (2017). MOVES-Matrix and Distributed Computing for Microscale Line Source Dispersion Analysis. Journal of the Air & Waste Management Association. <http://dx.doi.org/10.1080/10962247.2017.1287788>.**
- **Guensler, R., H. Liu, Y. Xu, A. Akanser, D. Kim, M. Hunter and M.O. Rodgers (2017). Energy Consumption and Emission Modeling of Individual Vehicles Using MOVES-Matrix. Transportation Research Record, No. 2627, pp. 93-102.**

# Questions:

- **Hongyu Lu**  
**LHY@gatech.edu**
- **Randall Guensler**  
**randall.guensler@ce.gatech.edu**
- **Please send questions by e-mail to both of us**