User's Guide for the Vertical City Weather Generator (VCWG v1.5.0)

Amir A. Aliabadi and Xuan Chen
Atmospheric Innovations Research (AIR) Laboratory
School of Engineering, University of Guelph, Guelph, Canada
http://www.aaa-scientists.com
aliabadi@uoguelph.ca

This document is typeset using LaTeX

September 26, 2025

1 About VCWG

The Vertical City Weather Generator (VCWG) is a software that predicts the urban micro-climate and building performance variables in relation to a nearby rural climate given the urban characteristics. VCWG predicts vertical profiles of temperature, wind speed, humidity, and turbulence kinetic energy as well as the building cooling/heating loads and natural gas/electricity consumption. More details on the model can be found at the Atmospheric Innovations Research (AIR) laboratory website at www.aaa-scientists.com and corresponding publications [Moradi et al., 2021, Moradi, 2021, Aliabadi et al., 2021, Moradi et al., 2022]. VCWG v1.5.0 is similar to VCWG v1.4.5 and v1.4.6, with the additional capability that it can allow optimization studies, using a micro-genetic algorithm developed by [Park and Park, 2021]. This guide accompanies the publication by [Aliabadi et al., 2023].

2 Basic Setup

2.1 Setting the Climate Forcing Files

To run the VCWG, it is required to put the weather file (*.epw) of the region of interest in the directory e.g. "/resources/epw/ERA5_Toronto_2020.epw". This file can be downloaded from EnergyPlus (`https://energyplus.net/`) or prepared using alternative datasets. In the released version of the software it is prepared using the ERA5 data product from the European Centre for Medium-Range Weather Forecasts (ECMWF) [Aliabadi and McLeod, 2023].

2.2 Setting Input Parameters

VCWG can take input parameters from the files located in the directory "/resources/parameters/". These files contain the required parameters of the case study including urban characteristics, vegetation parameters, view factors, simulation parameters, and building renewable and alternative energy configurations. In the released version of the software there are 13 input parameter files for Toronto. File "initialize_Toronto_0.uwg" is associated with a case with *all* renewable energy options and is set up to run the model for 31 days in January. To consider the renewable energy options for the model, variable "Adv_ene_heat_mode" should be set. To include only photovoltaic panels and wind turbine, this parameter should be set to 3. For inclusion of other renewable energy options (solar thermal, building integrated thermal energy storage, phase change materials, and energy recovery), this variable should be set to 1 for heating mode and 0 for cooling mode of operation.

Files "initialize_Toronto_1.uwg" to "initialize_Toronto_12.uwg" are used for annual runs of VCWG, and they are set up to run for each month of the year. Lines 230, 231, 232 contain the starting month, starting day, and the duration of the simulation in number of days. For each month, it is desired to start the simulation 3 days before the start of the month and then discard the first 3 days of data as spin up data. For example, to simulate February, one can start from Month = 1 (January), Day = 29 (3 days before start of February), for nDay = 31 (28 days in February plus 3 days for spin up). The January simulation is an exception because the EPW files only contain data for single year. The user is able to change the parameters to define and run a simulation of interest.

There are three more files in which input parameter specifications should be made. 1) To change the building envelop properties (e.g. resistance values) and HVAC equipment specification (e.g. coefficient of performance and thermal efficiency) the file "LocationSummary" in directories "/resources/DOERefBuildings/BLD1" through "/resources/DOERefBuildings/BLD16" should be modified corresponding to building type. Note that in the released version of the software the appropriate folder is "/resources/DOERefBuildings/BLD6" for mid rise apartment. Also note that only the last column of the spreadsheet is advised to be modified corresponding to custom values. i.e. the users are discouraged to change values in the other columns. 2) To change the ventilation and infiltration rates, the appropriate file "ZoneSummary" in directories "/resources/DOERefBuildings/BLD1" through "/resources/DOERefBuildings/BLD16" should be modified corresponding to building type. Note that in the released version of the software the appropriate folder is specified for the mid rise apartment as "/resources/DOERefBuildings/BLD6".

Two copies of the original files "BLD6_LocationSummaryOriginal" and "BLD6_ZoneSummaryOriginal" are kept in directory "/resources/DOERefBuildings/BLD6". As will be explained later, values in files "BLD6_LocationSummary" and "BLD6_ZoneSummary" may be changed frequently, so it is a good idea to keep a fresh copy of the original values.

If desired, new view factors can be obtained by running "/UWG/Run_RayTracing.py" and copy and paste the results from file e.g. "/UWG/ViewFactor_Toronto.txt" into the input file e.g. "/resources/parameters/initialize_Toronto_0.uwg" or other initialization files.

3 Running VCWG for Basic Simulations

There are two options for running a basic simulation of VCWG v1.5.0: the single and serial modes. The single mode only runs the model given one set of input parameters, while the serial mode allows running the model for 12 consecutive months, requiring 12 initialization files. For the single mode, in the python file "/VCWG/VCWGv1.5.0.py" located in the main directory, the user is required to change the name of weather file and the name of the initialization file to the ones located in the directories of "/resources/epw" and "/resources/parameters/", respectively. This run produces hourly results that are saved in directory "/Output/". For the serial mode, in the python file "/VCWG/VCWGv1.5.0Serial.py" located in the main directory, the user is required to specify 1 weather file and 12 input parameter files for each month of the year. In addition, the user should specify the 12 file names for the output files for building performance metrics for the entire month to be saved in directory "/Output/". In either simulation mode, it takes a few minutes to generate the output files located in the "/Output/" directory. It is recommended to discard the first 72 hours (3 days) of simulation for each month as spin-up period while considering results after this period.

"VCWGv1.5.0.py" and "VCWGv1.5.0Serial.py" are designed to run on Python 2.7.13. This version of Python can be downloaded from `https://www.python.org/downloads/release/python-2713/`. For example, for a 64-bit Windows operating system the installation file will be "python-2.7.13.amd64". The following packages and versions can be used: numpy 1.14.3, scipy 1.1.0, matplotlib 2.2.2. Note that other packages may also work. "UWG/Run_RayTracing.py" is designed to run on Python 3.6.1. This version of Python can be downloaded from the following link `https://www.python.org/downloads/release/python-361/`. For example for a 64-bit Windows operating system the installation file will be "python-3.6.1-amd64". The following packages and versions can be used: numpy 1.19.5, scipy 1.1.0, matplotlib 3.1.1. Note that other packages may also work.

4 Running VCWG for Optimization Studies

To run VCWG for micro-genetic optimization, various other settings are necessary. In the associated file "opt_VariableSpace.py", the user specifies the range for all variables that may be optimized. In addition the increment for change of each variable must be set. That is to say that the micro-genetic optimizer does not optimize a variable in a continuous scale, but rather in a discrete scale. For example, the lower value for photovoltaic coverage is set to "lPV=0.1" per building footprint area [m^2 m^{-2}], the higher value for photovoltaic coverage is set to "hPV=0.6" per building footprint area [m^2 m^{-2}], and solutions with increments of 0.1 per building footprint area [m^2 m^{-2}] are possible. Next, three options are available for optimization. These are specified by setting variable "ene_mode". When "ene_mode=3", up to 9 possible variables can be optimized. This mode is associated with choice of only PV and wind turbine as renewable energy. If a value of 1 is chosen for optimization of a variable, then it is optimized; if zero is chosen, then it is not optimized. When "ene_mode=2", up to 7 possible variables can be optimized. This mode is associated with choice of no renewable energy, in which case, only basic building variables will be optimized. When "ene_mode=None", up to 15 possible variables can be optimized. This case is associated with utilization of most possible renewable energy options. In this case the advanced energy mode could be either 1 (for heating) or 0 (for cooling) depending on the month of analysis.

The micro-genetic optimizer needs to update the file "LocationSummary.csv" with new values of building thermal resistance for roof and walls. This is achieved by overwriting the file "LocationSummary.csv" after inner and outer iterations of the optimizer. Before the simulation begins, it is desirable to copy and rename the original file "BLD6_LocationSummaryOriginal.csv" into "BLD6_LocationSummary.csv". After few iterations of the code the values inside this file will change. The file "opt_UpdateCSV.py" is responsible for overwriting the "LocationSummary.csv" file.

The EPW file and the initialization files are defined inside the file "VCWG150Optimization.py". For example, in this distribution of the software the EPW file name is "ERA5_Toronto_2020.epw" and the initialization files range from "initialize_Toronto_1.uwg" to "initialize_Toronto_12.uwg" for each month. The "ene_mode" should be defined for each month in this file, to be just like the "Adv_ene_heat_mode" for each month in the input initialization files. The case name is formatted to include the outer iteration number "OuterIter", the inner iteration number "InnerIter", and the individual number in the population "Idv". As the optimizer progresses, it creates buildings with different attributes, each of which are assigned three numbers for identification (i.e. outer interation, inner iteration, and individual numbers).

The rest of the settings should be in file "opt_Run.py". This file has 5 main parts: in part 1, the "buildingtype=6" is currently set to mid-rise apartment. For the first VCWG run "base_ene_mode=2", which means no renewable energy is considered for the first run. The code takes none-zero and none-negative "MinTotalNetElecDemand = 10" and "MinTotalNetGasConsump = 20" to ensure normalization of the objective function is successful.

In part 2, the user runs VCWG once for a base case that should not include retrofits, renewable/alternative energy. The desired variables in each of the 12 initialization files should be set. Also the correct starting building envelop thermal resistance value should be set in the "LocationSummary.csv" file. One way is to recover the correct starting value from the original file "LocationSummaryOriginal.csv". After the optimization starts, up to 13 variables will be updated and passed to the function "opt_VCWG".

In part 3, the optimization is set up. There are 4 objective functions: electricity consumption, gas consumption, cost, and greenhouse gas emissions. However, only three objective functions are factored into a weighted sum to be minimized by optimization. These are electricity consumption, gas consumption, and cost. The following variables are set: "wgt", "n_pop", "iMaxInner", "iMinInner", "iMaxOuter", "iMinOuter", "ConvergenceThresholdInner", "ConvergenceThresholdOuter".

The "ene_mode" is set, which should be consistent with the 12 input initialization files as well as the "VCWG150Optimization.py" file.

In part 4, the optimization is executed. The code completes inner and outer loop iterations and monitors the convergence criteria. It breaks out of the inner and outer loops either when maximum number of iterations are reached or when convergence criteria are met. Throughout the iteration process, the results are saved in text files or CSV files.

Currently in each inner loop iteration the solution or individual attributes (genes) are just mixed, while no new individuals are sampled randomly. Currently the sample individual population is 5. After each inner loop iteration, the best individual, i.e. that with lowest overall objective function value, is kept as the "elite individual". For the elite individual the genetic code of the individual is not changed for the next inner iteration. The worst individual, i.e. that with highest overall objective function value, is discarded. For example, consider the population below, where each individual has 6 attributes (genes). Each gene holds the value of the variables to be optimized (e.g. "V_bites", "m_dot_st_f", "A_ST", "V_pcm", "T_melt", "m_dot_he_st")

- Individual 1: a1 a2 a3 a4 a5 a6 (lowest objective function, elite)
- Individual 2: b1 b2 b3 b4 b5 b6
- Individual 3: c1 c2 c3 c4 c5 c6
- Individual 4: d1 d2 d3 d4 d5 d6
- Individual 5: e1 e2 e3 e4 e5 e6 (highest objective function, discard)

From the remaining 4 individuals (including the elite individual) four children are created by pairing two parents and mixing their genes. The genes are mixed by taking a sequence of individual attributes and exchanging them at an arbitrary point. For the individual sample above, we obtain

- Individual 1: a1 a2 a3 a4 a5 a6 (elite)
- Individual 2: a1 a2 b3 b4 b5 b6 (mixed genes)
- Individual 3: b1 b2 a3 a4 a5 a6 (mixed genes)
- Individual 4: c1 c2 c3 c4 d5 d6 (mixed genes)
- Individual 5: d1 d2 d3 d4 c5 c6 (mixed genes)

If the maximum number of inner iterations are reached, or if the convergence criteria for the inner iteration is satisfied, then the elite individual is kept, while 4 new individuals are randomly sampled to continue to the next inner iteration. For example, from the above sample, the following sample will be generated.

- Individual 1: a1 a2 a3 a4 a5 a6 (elite)
- Individual 2: f1 f2 f3 f4 f5 f6 (new individual)

- Individual 3: g1 g2 g3 g4 g5 g6 (new individual)
- Individual 4: h1 h2 h3 h4 h5 h6 (new individual)
- Individual 5: i1 i2 i3 i4 i5 i6 (new individual)

This process continues, until maximum number of outer iterations are reached, or the convergence criteria for the outer iteration is satisfied.

In part 5, the results are also saved in python array format using hyper dimensions. The optimization process takes a long time. It is better to run the code on a cluster using a single CPU. To speed up the simulation process, it is better to run simultaneous VCWG computations on multiple CPUs. Each CPU can complete simulation for a given configuration, location, year, etc.

References

[Aliabadi et al., 2023] Aliabadi, A. A., Chen, X., Yang, J., Madadizadeh, A., and Siddiqui, K. (2023). Retrofit optimization of building systems for future climates using an urban physics model. *Building and Environment*, 243:110655.

[Aliabadi and McLeod, 2023] Aliabadi, A. A. and McLeod, R. M. (2023). The Vatic Weather File Generator (VWFG v1.0.0). *Journal of Building Engineering*, 67:105966.

[Aliabadi et al., 2021] Aliabadi, A. A., Moradi, M., McLeod, R. M., Calder, D., and Dernovsek, R. (2021). How Much Building Renewable Energy Is Enough? The Vertical City Weather Generator (VCWG v1.4.4). *Atmosphere*, 12(7):882.

[Moradi, 2021] Moradi, M. (2021). *The Vertical City Weather Generator*. PhD thesis, University of Guelph, Guelph.

[Moradi et al., 2021] Moradi, M., Dyer, B., Nazem, A., Nambiar, M. K., Nahian, M. R., Bueno, B., Mackey, C., Vasanthakumar, S., Nazarian, N., Krayenhoff, E. S., Norford, L. K., and Aliabadi, A. A. (2021). The Vertical City Weather Generator (VCWG v1.3.2). *Geosci. Model Dev.*, 14(2):961–984.

[Moradi et al., 2022] Moradi, M., Krayenhoff, E. S., and Aliabadi, A. A. (2022). A comprehensive indoor–outdoor urban climate model with hydrology: The Vertical City Weather Generator (VCWG v2.0.0). *Building and Environment*, 207:108406.

[Park and Park, 2021] Park, S. and Park, S. K. (2021). A micro-genetic algorithm (GA v1.7.1a) for combinatorial optimization of physics parameterizations in the Weather Research and Forecasting model (v4.0.3) for quantitative precipitation forecast in Korea. *Geosci. Model Dev.*, 14(10):6241–6255.