

SkyGlow 6.2 - manual

The present version of SkyGlow is only for tests purposes and for those who want to become familiar with this tool. SkyGlow is not dedicated for commercial use, so any potential user should accept that the product features are partly limited. Our capacities for programming are restricted, but we appreciate any suggestions for improvements of the present version of SkyGlow. We strongly encourage those users who are interested in this product to send us comments that can help to improve next versions of SkyGlow.

GUI of SkyGlow is written in Java (it is platform independent), so you have to install the last version of the Java Runtime Environment or the last version of the Java Development Kit. Then you have simply to unpack the SkyGlow.zip file to any folder on your computer and run SkyGlow.jar program. However, SkyGlow uses GNUplot visualizer for the graphical presentation of results, and thus also GNUplot must be installed.

In comparison to older versions of the software, this version is parallelized (and much faster) and allows to find the total number of lumens (most important parameter for light pollution) produced by a city from VIIRS-DNB night composite image.

Contents

1. Installing and running the SkyGlow 6.2 software in Microsoft Windows	2
2. Basic settings of the SkyGlow software	2
3. Basic test of the SkyGlow software	4
4. Running the computation without GUI	6
5. Changing parameters of the model	7
Setting the name of the model and the description of the calculation	7
Including cities into calculation	7
Changing the city emission function.....	9
Setting the position of the observer.....	10
Changing the local parameters (atmosphere, horizon) and specifying the outputs.....	11
Displaying the results.....	13
Displaying relative changes	14
Splitting/enlarging the city	17
Setting the precision/resolution	19
6. Archiving the projects.....	20
7. Creating own model	20
Creating the city-files.....	21
Creating the spectra of lights	22
Mixing the spectra of lights	24
Creating the spectral reflectivity of clouds.....	24
Creating the horizon profiles.....	24
Creating the aerosol SSA and ASY files	25
8. Calibrating the city emission function using VIIRS-DNB data	25
Download of the VIIRS data file.....	25
Data extraction in QDIS	25
Export GeoTiff data into a standard text-file.....	27
Calibrate the total lumens of cities	28
9. Troubleshooting the problems	30

1. Installing and running the SkyGlow 6.2 software in Microsoft Windows

Program was successfully tested under Windows 10/11 (64-bit versions). Please, follow next steps:

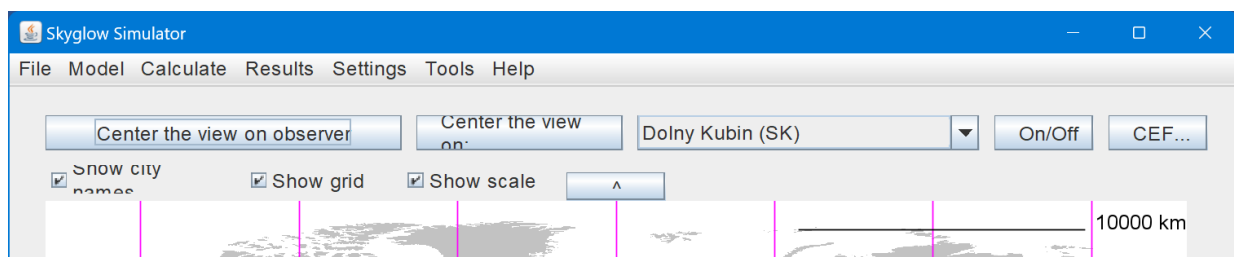
1. In the case you don't have GNUplot already installed: you can download version 6 and install it from the web-site
<http://sourceforge.net/projects/gnuplot/files/gnuplot/>
We have made successful tests with the current version 6.02 (**gp602-win64-mingw.exe**) and latest older version 5.4.10. Please, do not disable the association of *.PLT files with GNUplot during the installation. After correct installation, the GNUplot is started if any PLT-file is double-clicked.
2. In the case you don't have Java Runtime Environment already installed: you can download and install it from the web-site
<http://java.com/en/download/>
We have made successful test with the current Version 8 Update 441 and older versions of Java 8. After correct installation, Java is started if any JAR-file is double-clicked.
3. Unpack the **SkyGlow.zip** file: You can unzip the file to any folder in your computer where you have permissions to create the files (e.g. "My Documents" or "Desktop" folder or any subfolder). Afterwards you should run (double-click) the file
SkyGlow.jar

2. Basic settings of the SkyGlow software

Although Java is platform-independent, the use of fonts and screen resolutions is manifold on different platforms. Various computers can have various fonts installed and the look of java-programs can change.

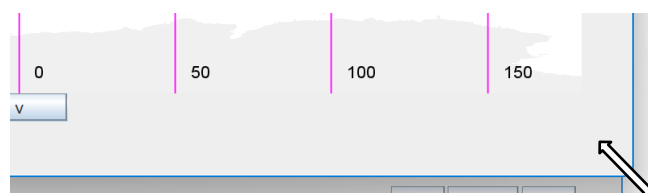
The font "Arial" is present by default in all versions of Microsoft Windows, so we designed our software for this font and the look of the SkyGlow software should be the same in all versions of Microsoft Windows.

But we cannot guarantee that these fonts are present in your computer. If the program looks like in following figure (texts are too wide and do not fit into reserved space, e.g. in buttons), you have to select another font from list of available fonts.



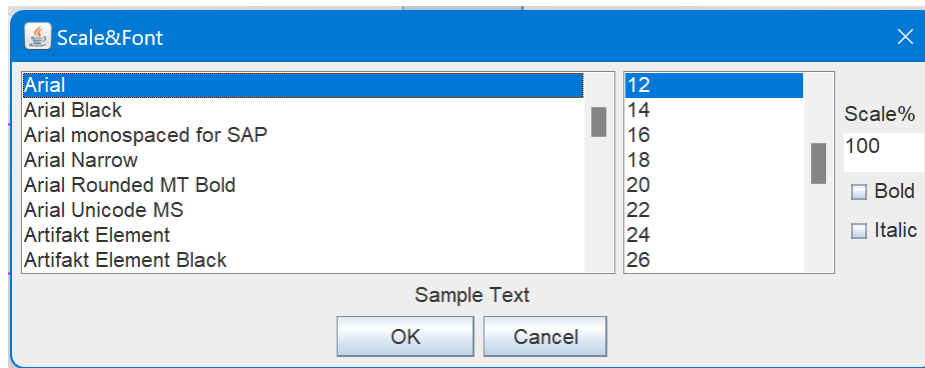
Improper font: texts are too wide and do not fit the reserved space (e.g. in buttons)

In first step, resize the SkyGlow-window to an appropriate size by dragging the right-bottom corner of the window:



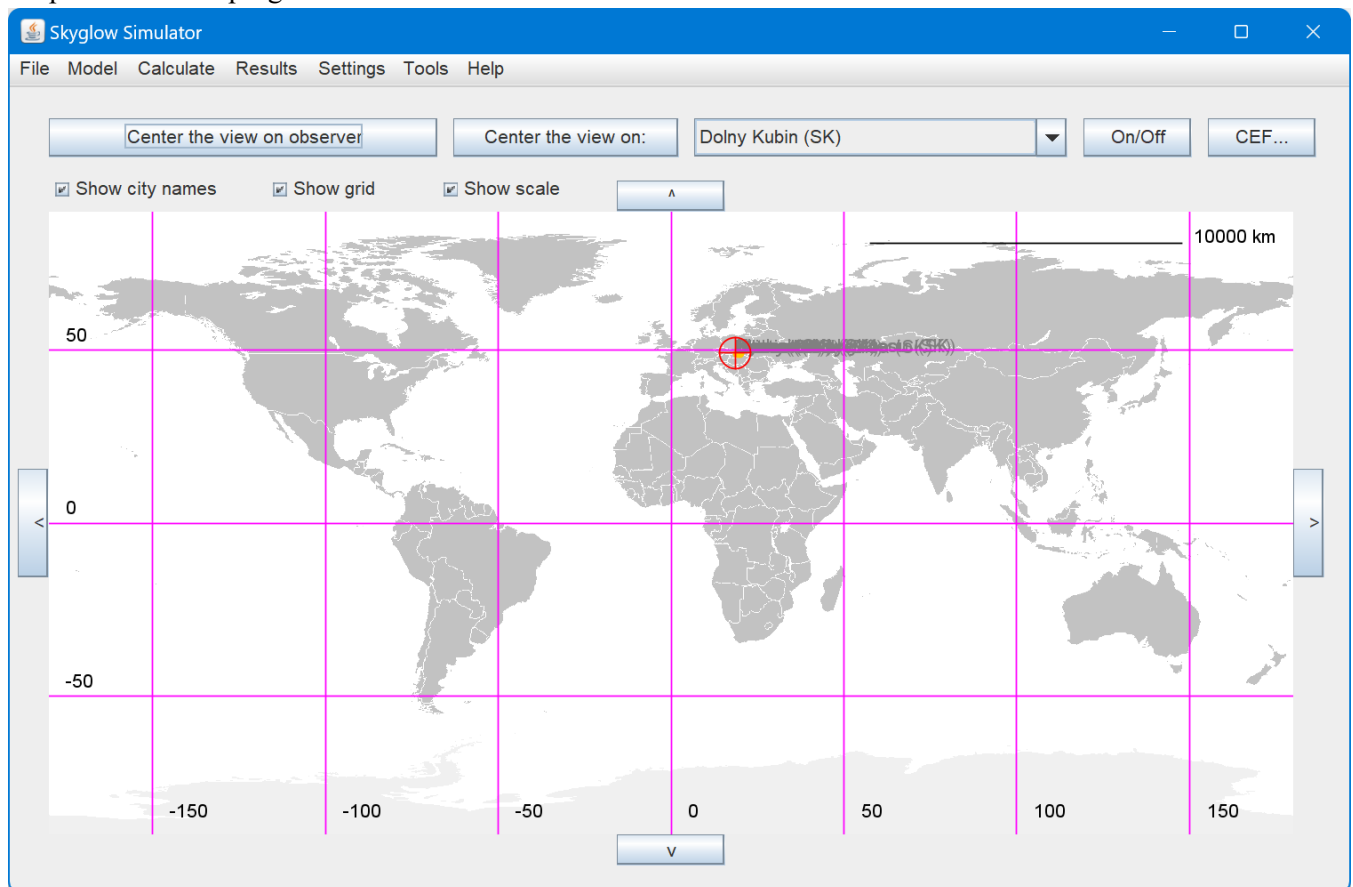
Changing the size of the application window

Then use the “Settings/Font & Application size” menu item and find a suitable font:



You can also experiment with the size of the font (usually 11 or 12 is the best). Try also “bold” or “italic” modifications. Several font families include also “Condensed” or “Narrow” modification of the font. Selected font is saved and will be used in future runs of the program. If needed, you can choose/save another font in any time. The “Scale%”-parameter controls the size of the application window.

Proper look of the program is as follows:



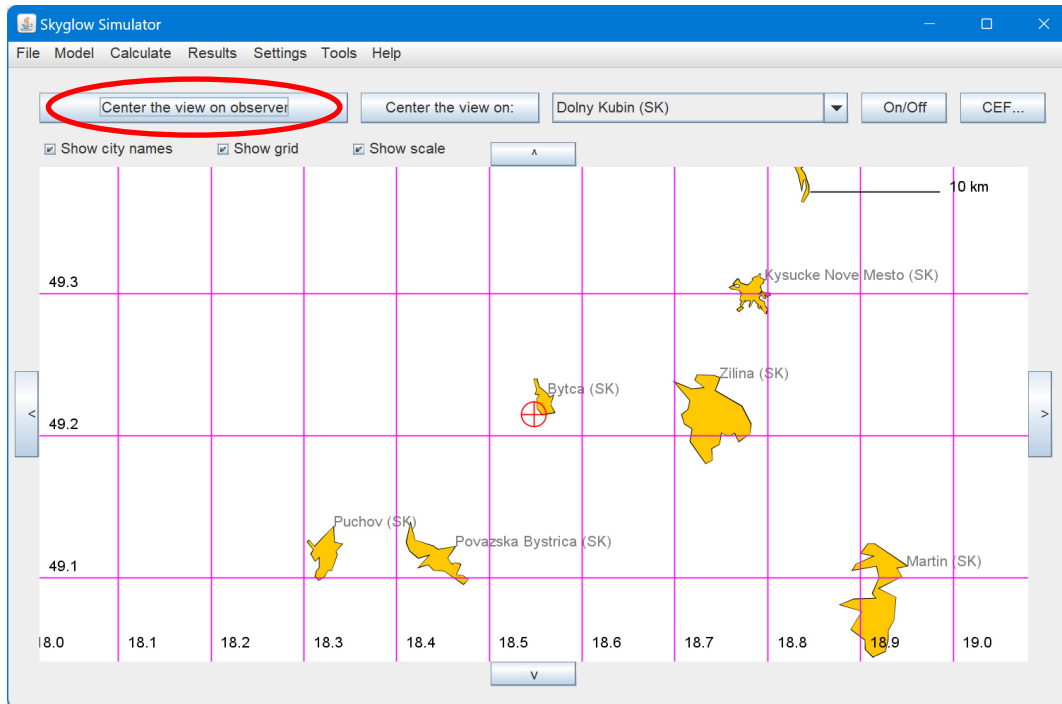
Proper look of the program

In the case you are not happy with the pixel resolution of the application (the content of the application window is sometimes not sharp on high-DPI screens), right-click the “SkyGlow.exe” launcher, go to submenu “Properties/Compatibility” and click the “Change high DPI settings”. In the dialog check the “Override high DPI scaling behavior. Scaling performed by: Application”. Click twice OK to apply the changes. Use the “SkyGlow.exe” launcher to start the SkyGlow software and resize the SkyGlow-window to appropriate size.

3. Basic test of the SkyGlow software

Create a new default project using 'File/New default project' menu item.

Click "Center the view on observer" button to see the cities close to the observer (red cross). You will see picture like below:



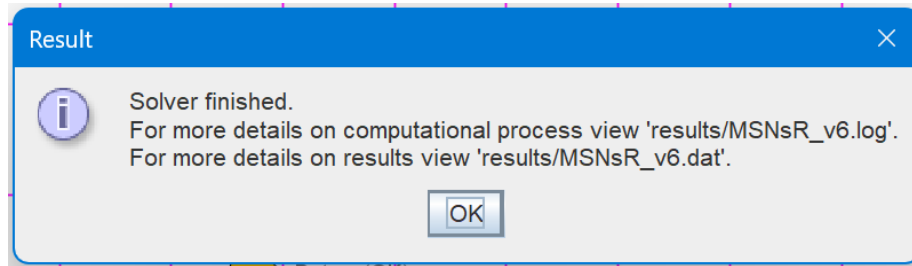
The default parameters of the model (city, sky) are used, if not changed by the user. So, we can start the calculations immediately.

Select the "Calculate/Save project & Solve" menu item. The solver (singleton written in C++) will start in a console:

```
C:\Users\frant\OneDrive - Univerzita Komenského v Bratislave\NewWork\SkyGlow_Fecko\GUI_v6.1\programs\windows\MSNsR_v6.exe
= 9.3403E-06, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.7s>
Integrating dome sample [200/3456]: azimuth: 275.3, zenith: 20.1; FirstPartOfEq30_H = 2.3075E-07, SecondPartOfEq30_H
= 9.5447E-06, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.6s>
Integrating dome sample [300/3456]: azimuth: 26.1, zenith: 27.2; FirstPartOfEq30_H = 2.4104E-07, SecondPartOfEq30_H
= 1.0382E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.5s>
Integrating dome sample [400/3456]: azimuth: 57.8, zenith: 30.7; FirstPartOfEq30_H = 2.4594E-07, SecondPartOfEq30_H
= 1.3205E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.4s>
Integrating dome sample [500/3456]: azimuth: 45.6, zenith: 34.2; FirstPartOfEq30_H = 2.4484E-07, SecondPartOfEq30_H
= 1.3256E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.3s>
Integrating dome sample [600/3456]: azimuth: 1.8, zenith: 37.8; FirstPartOfEq30_H = 2.3490E-07, SecondPartOfEq30_H
= 1.0753E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.2s>
Integrating dome sample [700/3456]: azimuth: 286.9, zenith: 37.8; FirstPartOfEq30_H = 2.2095E-07, SecondPartOfEq30_H
= 1.3131E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.1s>
Integrating dome sample [800/3456]: azimuth: 194.1, zenith: 40.1; FirstPartOfEq30_H = 2.2992E-07, SecondPartOfEq30_H
= 1.1006E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.1s>
Integrating dome sample [900/3456]: azimuth: 87.6, zenith: 43.7; FirstPartOfEq30_H = 2.5162E-07, SecondPartOfEq30_H
= 2.3834E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 22.0s>
Integrating dome sample [1000/3456]: azimuth: 327.6, zenith: 43.7; FirstPartOfEq30_H = 2.2323E-07, SecondPartOfEq30_H
= 1.2456E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 21.9s>
Integrating dome sample [1100/3456]: azimuth: 192.0, zenith: 47.2; FirstPartOfEq30_H = 2.2704E-07, SecondPartOfEq30_H
= 1.2325E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 21.9s>
Integrating dome sample [1200/3456]: azimuth: 50.5, zenith: 51.9; FirstPartOfEq30_H = 2.4825E-07, SecondPartOfEq30_H
= 2.3463E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 21.8s>
Integrating dome sample [1300/3456]: azimuth: 257.7, zenith: 51.9; FirstPartOfEq30_H = 2.0734E-07, SecondPartOfEq30_H
= 1.9316E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 21.7s>
Integrating dome sample [1400/3456]: azimuth: 97.8, zenith: 54.3; FirstPartOfEq30_H = 2.5507E-07, SecondPartOfEq30_H
= 4.2036E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 21.7s>
Integrating dome sample [1500/3456]: azimuth: 291.1, zenith: 55.5; FirstPartOfEq30_H = 2.0363E-07, SecondPartOfEq30_H
= 2.0897E-05, SecondPartOfEq30_TOP = 0.0000E+00 <ETA: 21.6s>
```

Please, note the ETA value (the estimated time to finish the job).

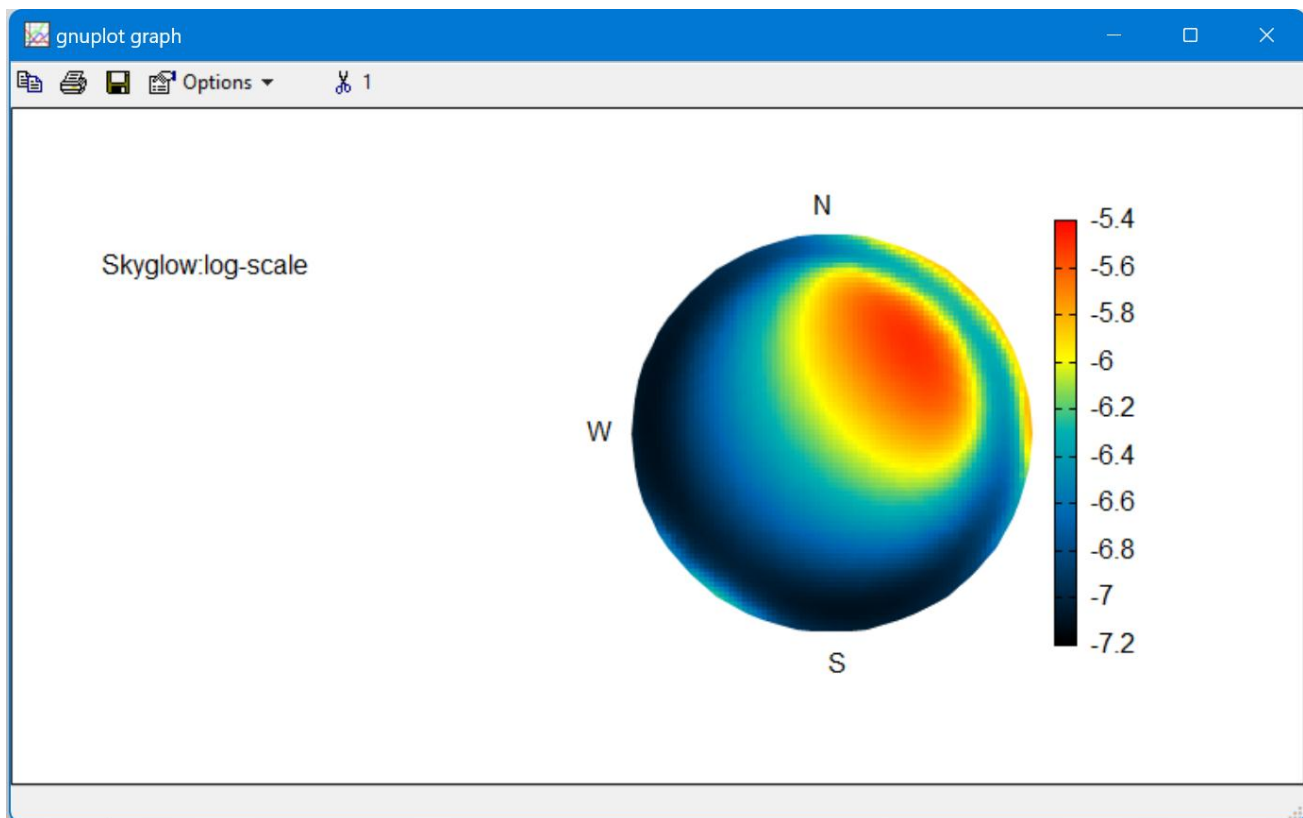
The following message is displayed after the solver has finished its job:



As you can read, more details about computations can be found in the file “MSNsR_v6.log” located in “results” folder. More results (computed values of the skyglow and main parameters used) are located in “MSNsR_v6.dat” file.

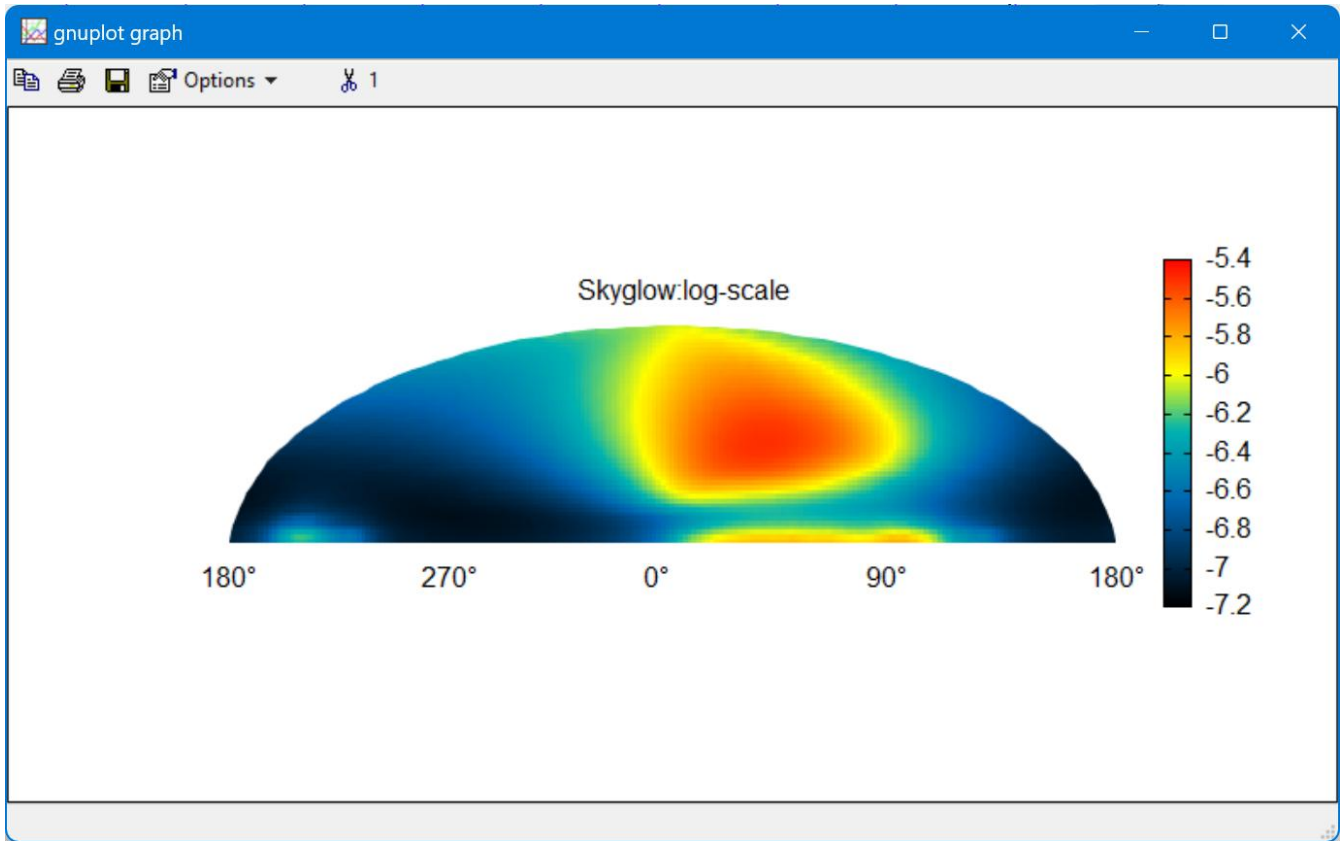
Select OK.

Then select the "**Results/Polar plot (God’s view)**" menu item. The GNUplot will be started and you will see figure like below. This figure represents calculated skyglow (logarithmic value) in polar-plot (God’s view). Such a representation is very suitable for the comparison of the calculated pattern with the map of cities.



Skyglow in polar-plot (God’s view)

Then select the "**Results/Hammer projection (Observer’s view)**" menu item. This second figure represents calculated skyglow (logarithmic value) in Hammer-projection (Observer’s view). Such a projection shows more details in the parts of the sky close to the horizon and less details of the sky close to the zenith (where usually only small changes are observed).



Skyglow in Hammer-projection (Observer's view)

Portable Network Graphics (PNG) versions of these pictures shown by GNUplot can be found in the "results" folder.

Computational details are printed to the "results/MSNsR_v5.dat" file. This file can be shown using "Results/More results" menu item. The calculation logs are stored in "results/MSNsR_v5.log" file. This file can be shown using "Results/Log-file" menu item.

All parameters of the calculation are stored in "results/MSNsR_v6.cfg" file. You can load these parameters using "File/Load last saved default project" menu item.

4. Running the computation without GUI

The calculation (eventually after changing the parameters in the CFG file) can be re-run without using GUI by double-clicking the "results/run.cmd" file.

If the CFG-file was changed and the calculation was re-run without GUI, run "plot_convert_data.cmd" to convert data for polar and Hammer plots. Then run "plot_hammer.cmd" and "plot_polar.cmd" to re-create plots.

The PNG files can also be re-created by double clicking corresponding PLT files.

5. Changing parameters of the model

The GUI allows you to make following changes in the computational model:

- set the name of the model and the description of the calculation
- switch any city on/off (include/exclude it from the calculation)
- change the parameters of the city emission function (e.g. spectrum or radiant intensity as a function of zenith angle) and other parameters of the city selected
- use a clear sky model or various overcast conditions (including partly cloudy condition)
- change the position of the observer and the direction of observer's view in Hammer projection
- change location-specific parameters (aerosol content and type, ...)
- change the output spectral band or switch between the output quantities: photometric or radiometric
- split or enlarge the city to simulate the city reconstruction

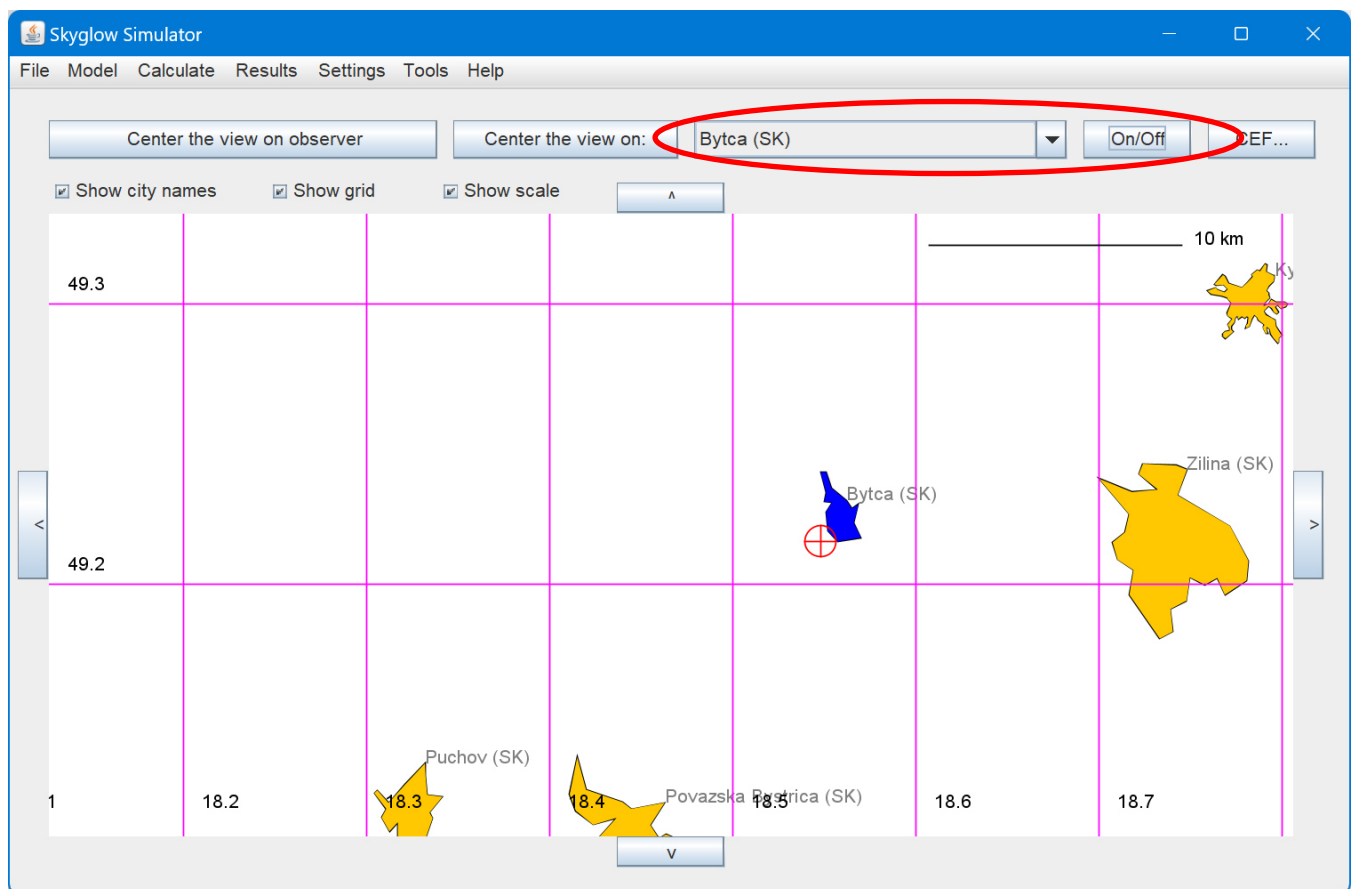
To set the parameters of the model, select the corresponding “Model”-menu item or use the mouse (see “Help/Quick mouse actions” for details).

Setting the name of the model and the description of the calculation

Use menu “Model/Description” to change the descriptions. Both are stored in “results/MSNsR_v6.cfg” file.

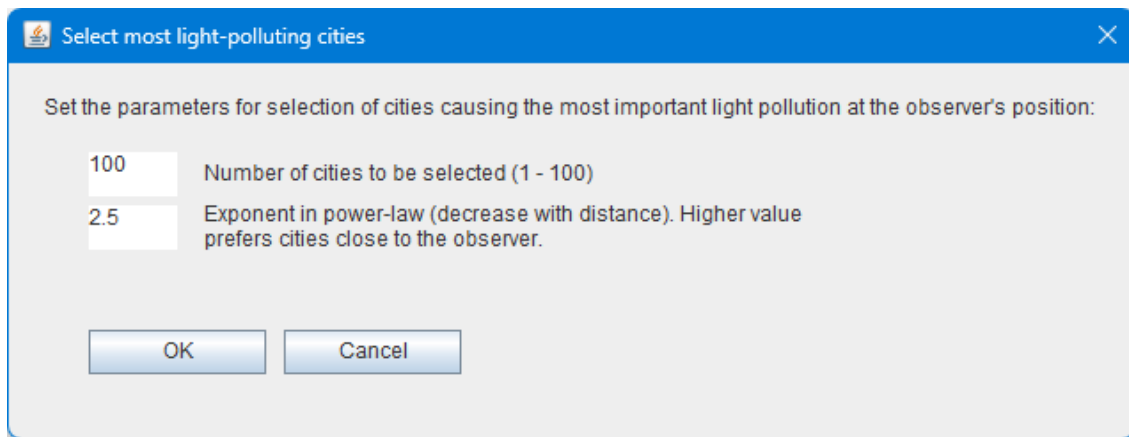
Including cities into calculation

To include/exclude an individual city from the model, **left-click the city** (use mouse wheel to zoom in/out the map). The city will change its color from orange (included) to blue (excluded) and vice-versa.



Alternatively, select corresponding city in the combo-box and click the “On/Off” button. Menu items “Model/Include all cities” and “Model/Exclude all cities” allow to include/exclude all cities at once.

In the case you have a big number of cities in your project, you can select only those that are expected to have the biggest influence on the skyglow at the observer's position. Go to the menu 'Model/Include most important cities...' and set parameters in the dialog window:



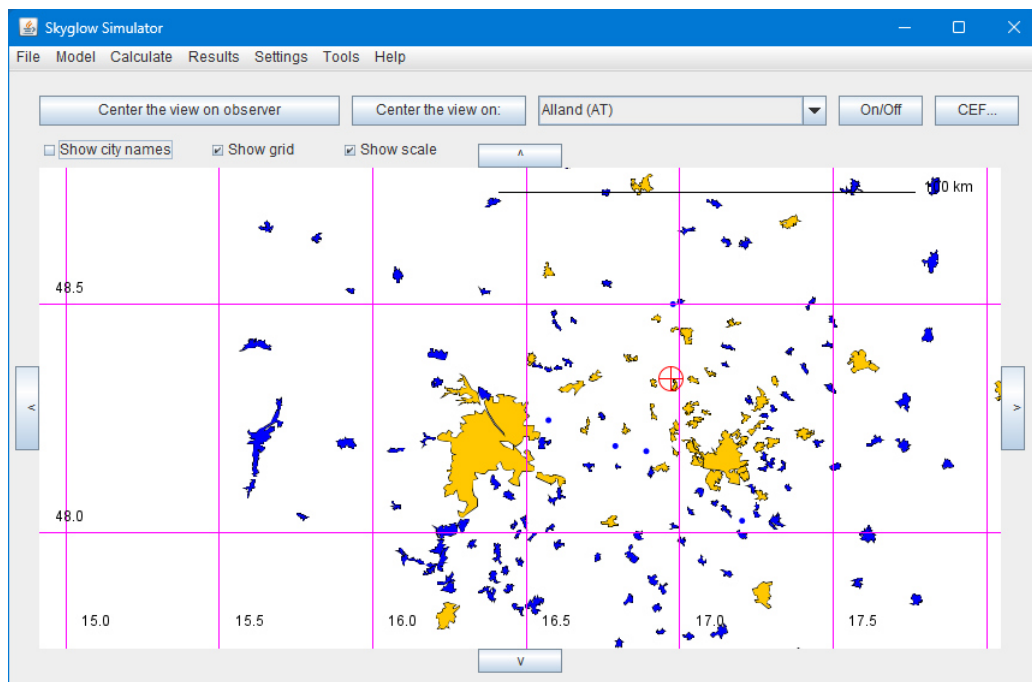
You can specify two parameters. The first one is the number of most important cities to be included into calculation. Selecting only a few cities leads to less accurate results, but to the short calculation time. This is good for quick estimation of sky pattern and light pollution levels.

The second parameter is exponent in the selection criterion:

$$criterion = \frac{\Phi}{r^n}$$

where Φ is the total lumens produced by the city, r is the distance of the city from the observer and n is the exponent in power-law describing the decrease of the skyglow with the distance. The value of 2.5 is typical for clear sky conditions, higher value prefers the cities closer to the observer and vice versa. Of course, the total number of lumens of the cities must be specified, or the population of the cities and their lumens per habitant must be set.

In the next picture, you can see the result of the selection of 30 most important cities (exponent = 2.5) for the Wien-Bratislava region. Observer's position is marked by red cross.



Changing the city emission function

To change the parameters of any city, **right-click** it. Alternatively, select the city in the **combo-box** and click the **“CEF...”-button**. A dialog will be displayed:

The dialog box is titled "City emission function - Zilina" and has three tabs: "Total emission", "Angular dependence", and "Spectral dependence". The "Total emission" tab is selected. It contains three radio buttons: "Total emission in watts (W)", "Total emission in lumens (lm)", and "Based on the city population (typically 500-1000 lm / habitant)". The third option is selected. To the right of the radio buttons are two input fields: the first contains "0.000e+00" and is labeled "Watts or Lumens", and the second contains "750.0" and is labeled "lumens per habitant". Below these is a third input field containing "83900" and labeled "is the city population". At the bottom are "OK" and "Cancel" buttons.

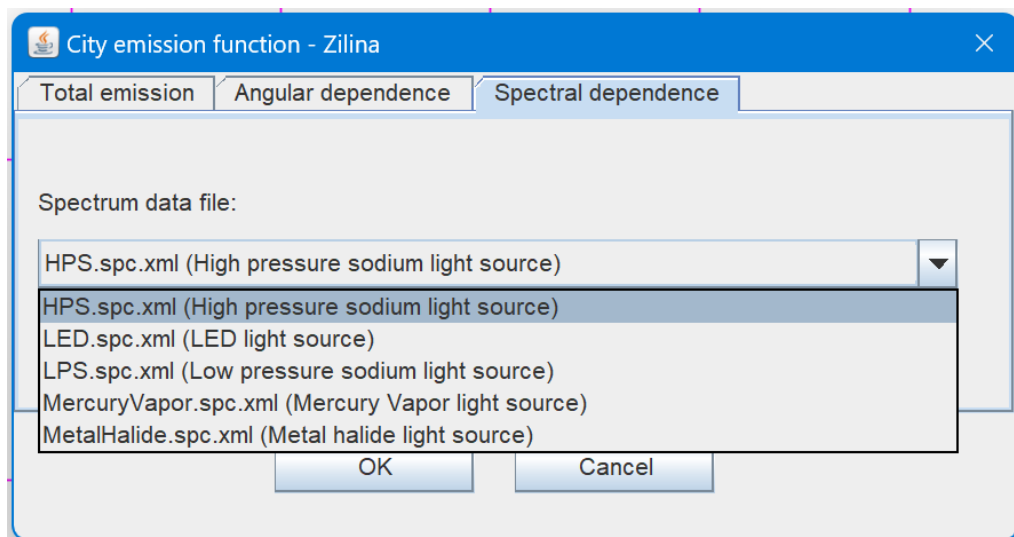
In the “Total emission” tab you can specify the total emission of the city in Watts or in lumens. The program automatically recalculates Watts to lumens (or back) using known spectrum of the light sources and using human-eye scotopic sensitivity curve. Eventually, you can estimate the emission using the city population and the typical number of lumens per habitant in your locality.

In the “Angular dependence” tab you can specify the angular pattern of the city emission function. We use two-parametric axisymmetric Garstang’s function. Two parameters that can be specified are: surface isotropic reflectivity and the fraction of light directly radiated toward the sky. Set this second parameter to zero and the city will be modelled as a cosine-source. Current version of the program does not support angular dependence in the form of a data-table.

The dialog box is titled "City emission function - Zilina" and has three tabs: "Total emission", "Angular dependence", and "Spectral dependence". The "Angular dependence" tab is selected. It contains two radio buttons: "Garstang" and "Data file (not implemented yet)". The "Garstang" option is selected. To the right of the radio buttons are two input fields: the first contains "0.15" and is labeled "Fraction of the light isotropically reflected", and the second contains "0.15" and is labeled "Fraction of the light radiated directly upwards". Below these is a third input field with a dropdown arrow. At the bottom are "OK" and "Cancel" buttons.

In the “Spectral dependence” tab you can select one typical spectra from the prepared set.

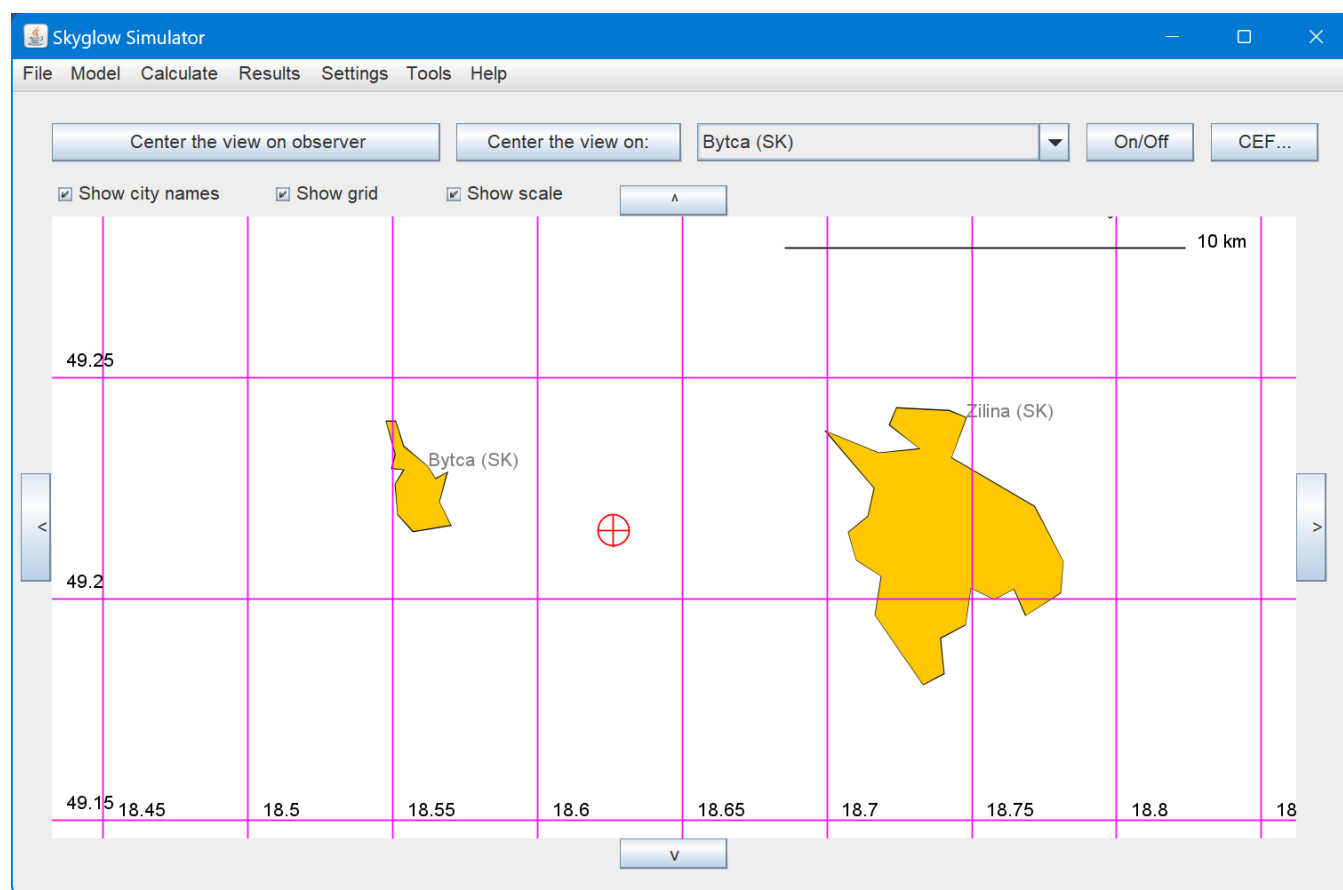
You can change CEF-parameters of a group of cities in one step. Select ‘Tools/Set emission parameters for a group of cities...’ menu item, select corresponding checkbox and filled-in value will be set to all cities currently showed in map (you will be prompted to confirm the change for every single city (or for all in once)).



You can create new city-file(s) using currently set parameters using “Tools/Update city-files”. Original city-files will be backed-up. You will be prompted to confirm every city-file to be updated.

Setting the position of the observer

To set the position of the observer **left-double-click** on any place in the map, e.g.:



If the exact GPS position of the observer is known, enter its **latitude and longitude** using “Model/Set observer’s position” menu item.

Changing the local parameters (atmosphere, horizon) and specifying the outputs

To change the local parameters **right-click** the map **outside any city**. The dialog will be displayed.

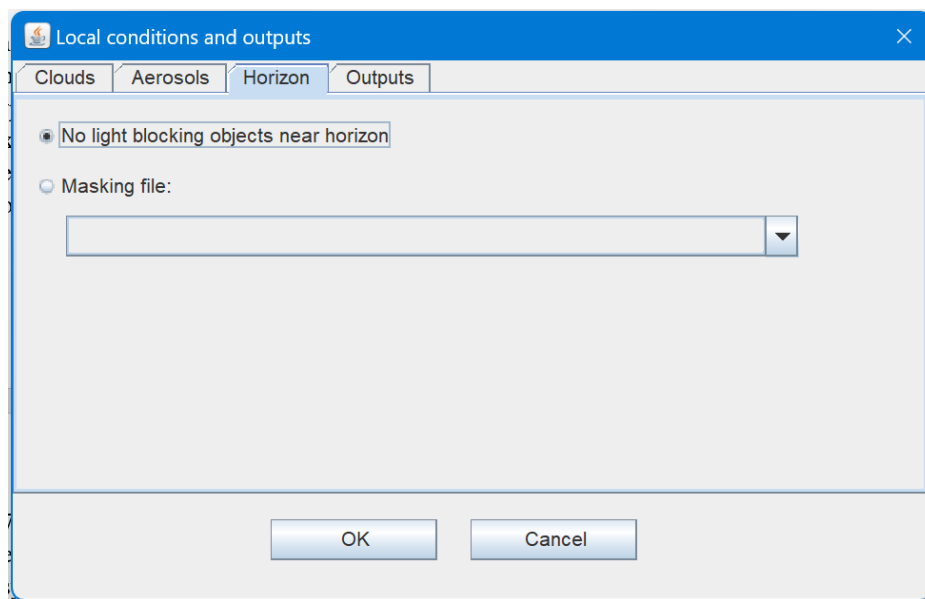
The screenshot shows the 'Local conditions and outputs' dialog box with the 'Clouds' tab selected. The 'Cloudless' radio button is unselected, and the 'Cloudy' radio button is selected. The 'Altitude of the cloud base [km]' is set to 1.0. The 'Cloud fraction [-]' is set to 0.3. The 'Render cloud field' checkbox is unselected. The 'Seed of random cloud pattern [-]' is set to 4. The 'Spectral albedo (select data file):' dropdown menu is set to 'Altocumulus.cld.xml (Altocumulus clouds)'. The 'OK' and 'Cancel' buttons are at the bottom.

In the “Clouds” tab you can specify cloudless or cloudy sky. The altitude of the cloud base in kilometers can be specified. The cloud fraction of 1 means totally cloudy sky. The cloud fraction less than 1 specifies partly cloudy condition. As an example, we can set the cloud fraction to 0.3. Clouds are generated at random position and size, the seed of random generator can be specified (seed is set to 4 as an example). Negative value of the seed means that the seed is determined by the computer time, so the cloud pattern differs in any calculation. You can check the “Render cloud field” checkbox to generate pictures of some views to the cloud layer. Finally, the spectral reflectance of clouds can be selected from prepared set.

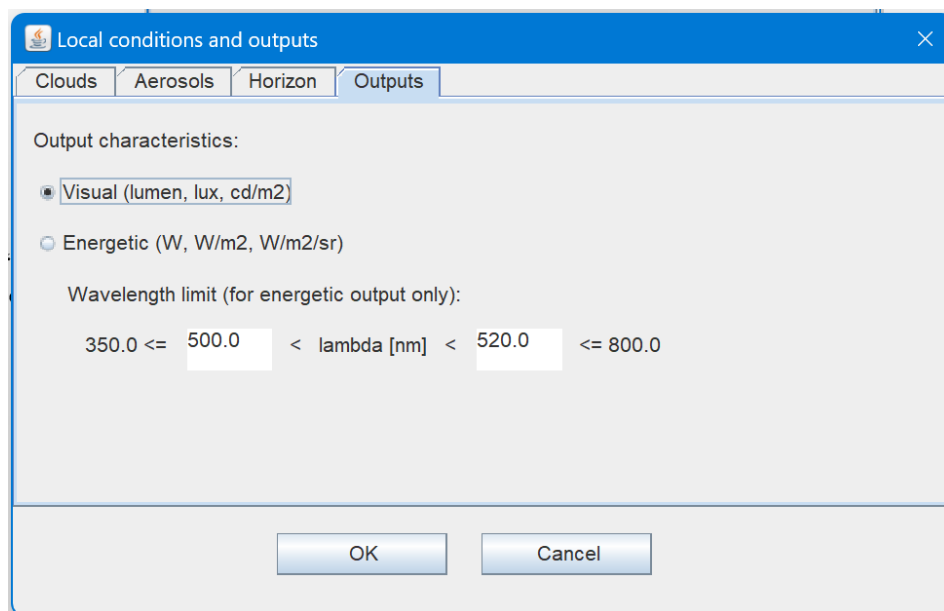
The screenshot shows the 'Local conditions and outputs' dialog box with the 'Aerosols' tab selected. The 'Reference aerosol optical thickness at 500 nm' is set to 0.1. The 'Angstrom exponent' is set to 1.3. The 'Scale height for the molecular atmosphere [km]' is set to 8.0. The 'Vertical gradient of the aerosol concentration [1/km]' is set to 0.65. The 'Data files for:' section has two dropdown menus: 'single scattering albedo:' set to 'constant_background.ssa.xml (Constant background)' and 'asymmetry parameter:' set to 'constant_background.asy.xml (Constant background)'. The 'OK' and 'Cancel' buttons are at the bottom.

In the “Aerosols” tab you can specify the parameters of aerosols. Reference aerosol thickness determines the amount of aerosol. Optical thickness less than 0.1 means very clear atmosphere, the value of 0.2 represents the typical not very turbid atmosphere, the value about 0.4 means very turbid (smoggy) atmosphere. Angstrom exponent determines the spectral dependence of the aerosol optical thickness. It depends on the type of the aerosol and its typical value is 1.1 – 1.5. Vertical gradient of the aerosol concentration determines how fast the concentration decreases with the height. The number represents the height where the aerosol concentration falls to

37% of its value at the Earth's surface. Similar parameter is the scale height of the molecular atmosphere and its very common value is 8 km. Finally, the spectral dependence of the single scattering albedo (SSA) and of the asymmetry parameter (ASY) is to be selected from prepared files. All the aerosol parameters needed in this tab are measured in AERONET-network stations located around the globe.



In the "Horizon" tab you can eventually select the horizon-profile at the observer's position (dependence of the elevation on the azimuth).

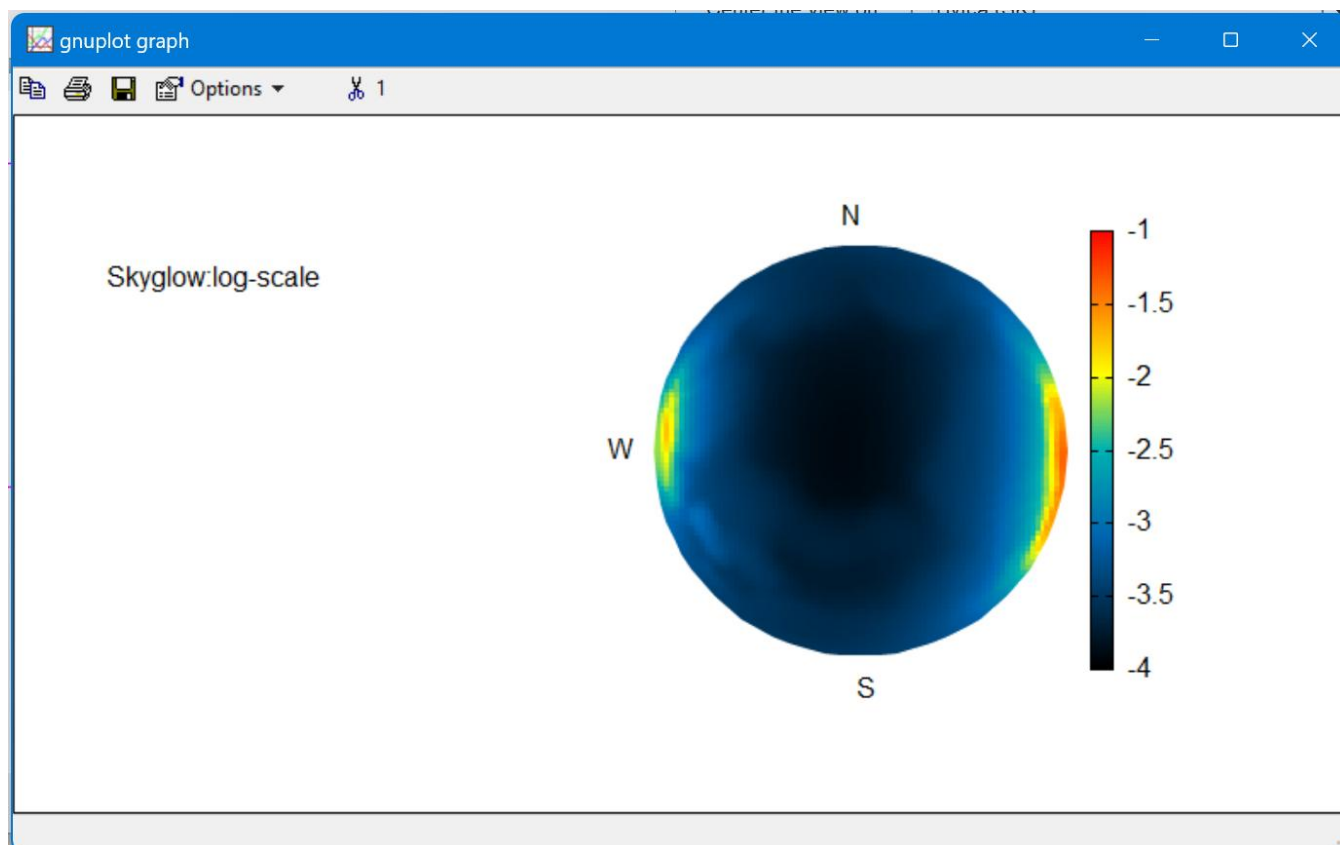
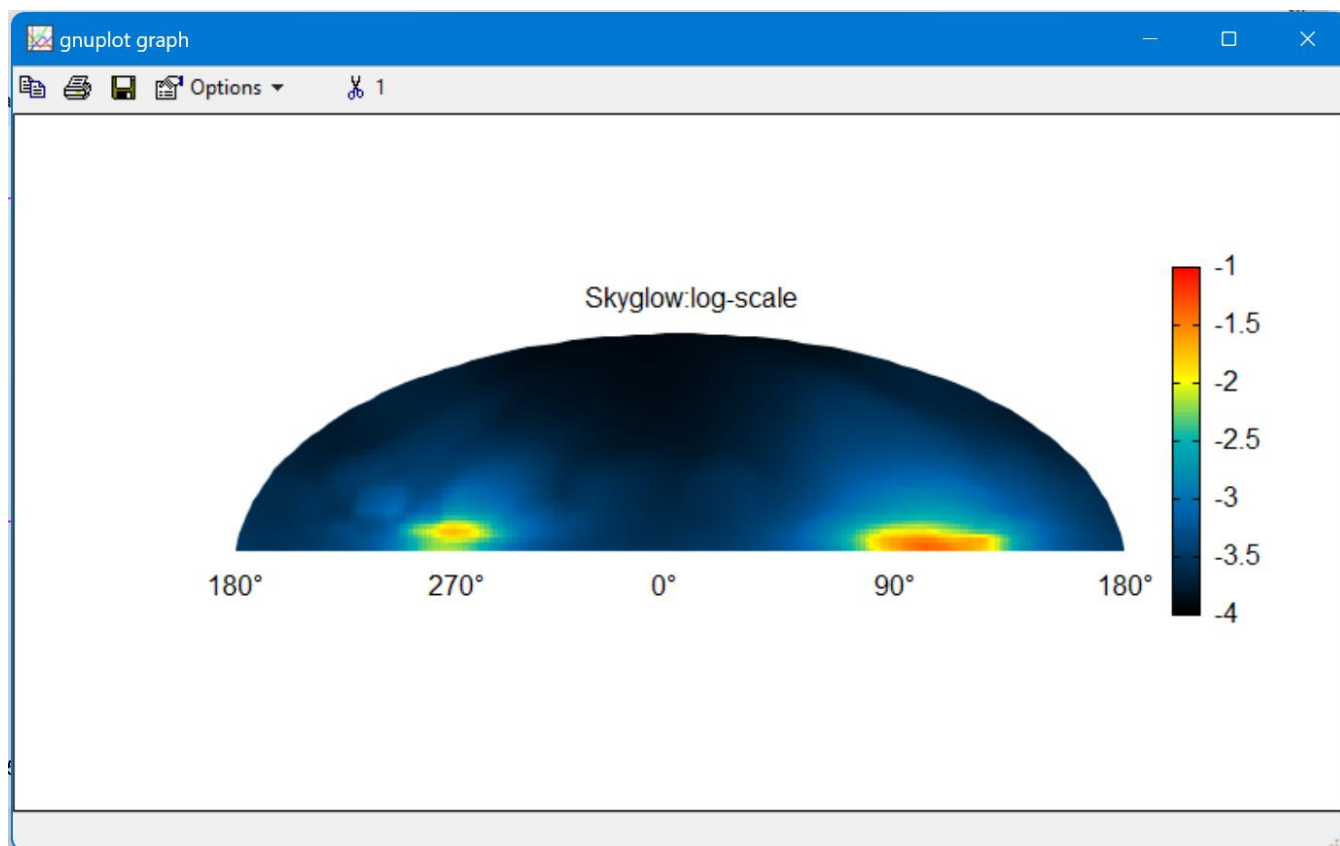


In the "Outputs" tab the output units can be specified. If "Visual" is selected (as is in the example), the skyglow in cd/m^2 is calculated. If "Energetic" is selected, the skyglow in $\text{W/m}^2/\text{sr}$ is calculated within the selected wavelength interval. By default, the wavelength-step of 5 nm is used in the program. If narrower interval is set, the skyglow is calculated at ends of the interval.

Alternatively, the local parameters can be set using "Model/Clouds...", "Model/Aerosols...", "Model/Horizon..." and "Calculate/Set output type..." menu items.

Displaying the results

After the solver is finished (use “Calculate/Save project & Solve” menu item), Select “Results/Polar plot (God’s view)” and “Results/Hammer projection (Observers’s view)” menu items. Our demo project has following results (in cd/m^2 , logarithm):



In Hammer projection you can specify the direction of observer's view (the azimuth of the center of the plot) in the "Results/Set observer's direction of view" menu item.

The appearance of the plots can be easily modified in corresponding PLT – files located in "results" folder. These files contain GNUplot commands and all commands are commented (explained). As an example, the section that defines the color palette looks like:

```
#--- following parameters significantly influence the colors of the plot -----
#--- uncomment (remove the # character at the beginning of the row) one of the pre-defined palettes ---
#
#set palette rgbformulae 7,5,15          #default color palette (black-blue-red-yellow)
#set palette rgbformulae 8,9,10         #color palette for B&W printing (blue-yellow)
#set palette rgbformulae 30,31,32       #another color palette for B&W printing (black-blue-magenta-
yellow-white)
#set palette defined (0 "black", 2 "blue", 4 "yellow", 6 "red") #color palette (black-blue-yellow-red)
set palette defined (0 0 0 0, 1.8 0 0.4 0.7, 3 0 0.7 0.7 , 4 1 1 0, 6 1 0 0) #color palette (black-cyan-
green-yellow-red)
#set palette gray negative              #inverted greyscale palette
#set palette gray                      #greyscale palette
```

By commenting/uncommenting the line the user can change the color palette. In another sections of the PLT – files user can: set fixed scale of the plots (cbrange), set/unset the isolines (cntrparam), set the font size and picture resolution (set terminal), change the size and position of the legend (colorbox), change numbers in the legend (clabel), set the text and the position of the title (set key), add any text at any position (set label), ... Use "Help/Plots formatting" menu item for more information.

There are two sets of PLT-files: "local settings" – located in "results" folder and "global settings" – located in "programs/windows" folder. Both sets are accessible also from "Results" menu. The "global settings"-files define the default appearance of the plots. The "local settings"-files are valid for current project and have higher priority. If local files are modified, the new appearance of plots will be applied. The appearance of plots in current project can be reset to "global settings" using "Results/Reset local plots settings" menu item.

Displaying relative changes

SkyGlow program allows you to view relative changes in skyglow after the project was modified (the spectrum of lights was changed or the city was enlarged or so on). The reference values can be loaded from any previously archived project (see Section 6).

Hint: It is strongly recommended to use the same resolution of project ('Calculate/Set resolution...' menu item) for both reference/current data. It means that the skyglow values are calculated in identical points of the sky. Otherwise the program applies the interpolation. However, the skyglow levels vary usually dramatically and such interpolation leads to artefacts and strange isolines, although basically the plot is correct.

How to display relative changes in skyglow:

1. Calculate the project and archive it (menu "File/Archive current project as..."). These results will be used as a reference later in step 4.
2. Return back to original project using "File/Load last saved default project" if you were working with default project. Eventually, use "File/Load archived project" if you were working with previously archived project.

Hint: The name of current folder (project) is displayed in the SkyGlow window title. This helps to be sure which project are you currently working with.

3. Make changes in your project and calculate the changed project.

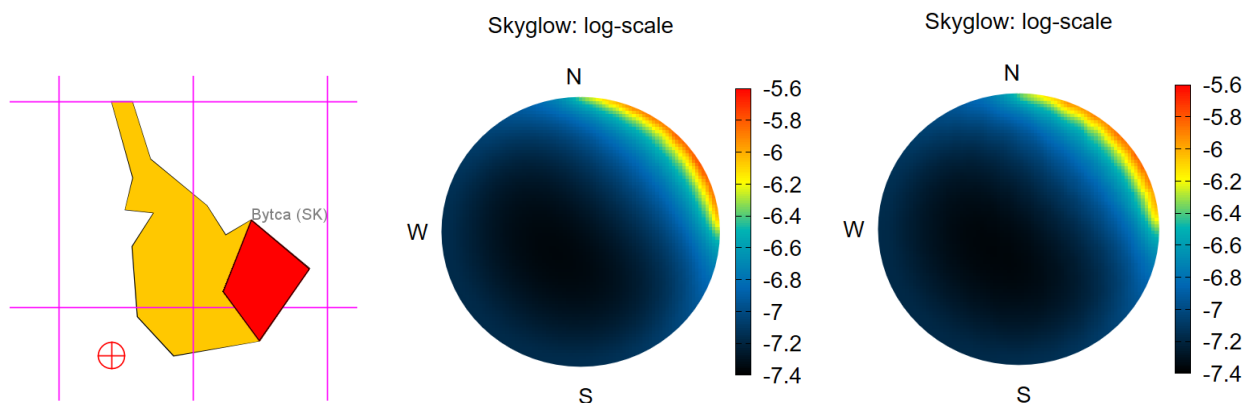
4. Use “Results/Select reference results...” menu item and select the project archived in step 1. In this step, file “MSNsR_v6.dat” (contains results) from archived project is copied into current project as “MSNsR_V6_reference.dat”. This file will be used permanently as a reference in current project, until you select another archived project as a reference.
5. Use “Results/Relative polar plot (God’s view)” and “Results/Relative Hammer projection (Observer’s view)” menu items to show relative plots. These plots display the value

$$\text{current_skyglow} / \text{reference_skyglow}$$

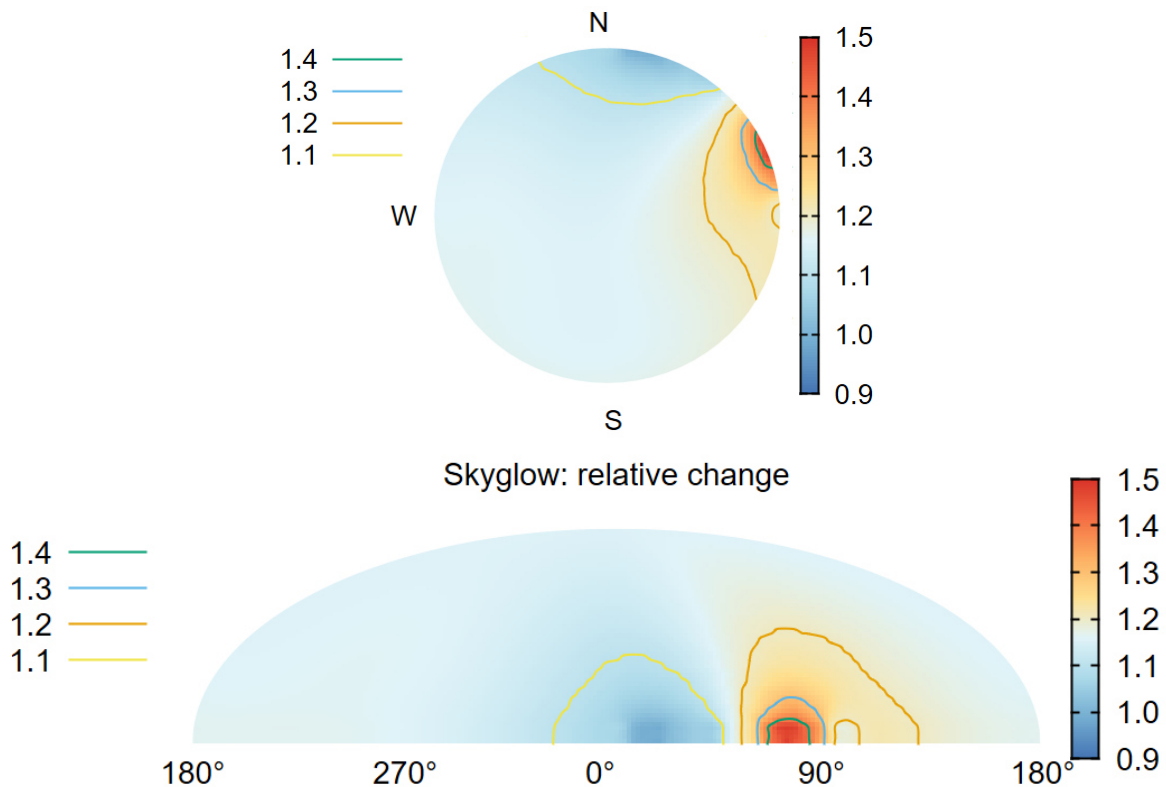
It means that value of 0.9 represents 10% decrease in skyglow, value of 1.5 means 50% increase in skyglow and so on.

Relative values are stored also in ‘**results_relative.dat**’-file in ‘results’-folder.

The relative plots are illustrated for the situation, where an industrial zone (red polygon) was built on the east side of the city (farther from the observer). In the middle/right is skyglow before/after the change. Clear sky conditions, green-colored light.



Typical relative plots look as follows:



The isolines can be set to any number and/or any value for highlighting threshold values. They can be changed by GNUplot command in PLT-files:

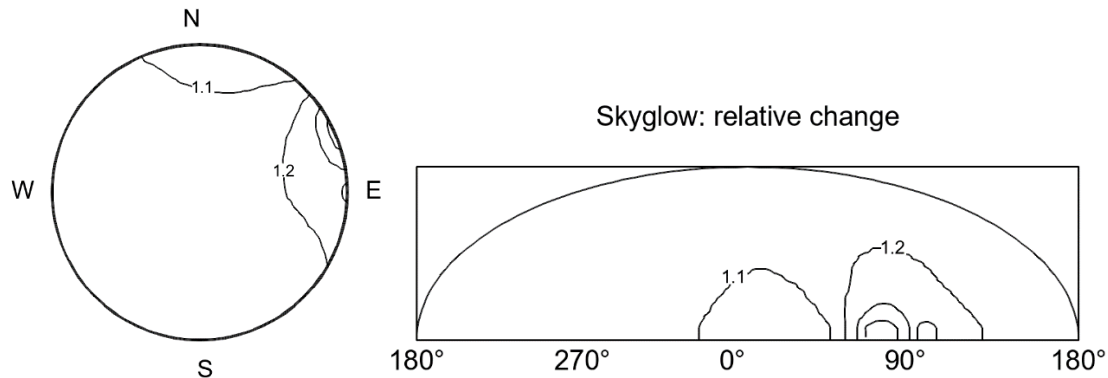
```
set cntrparam levels discrete 0.5,0.6,0.7,0.8,0.9,1.1,1.2,1.3,1.4,1.5 #exact isolines
```

In the case the relative values are spread over more orders of magnitude, you can try change the color scale to logarithmic by the command:

```
set logscale cb #colors will be in logarithmic scale
```

In our example can be seen that most of the sky (including zenith) became brighter by 10% - 20%.

Besides the color plots, the black & white plots with labeled isolines are saved to the disk (results_polar_relative_bw.png and results_hammer_relative_bw.png). They look like:

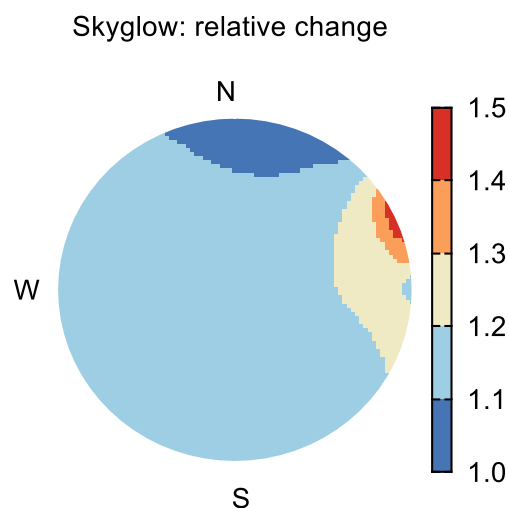


The position of labels and other parameters can be set by commands (at the end of PLT files):

```
set cntrparam point 1 #labels density on the isolines, use 1,2,3,...
```

```
set cntrlabel format '%.1f' font ',24' start 50 interval -1 #no. of decimal places, fontsize, starting position of labels on the isolines
```

A nice compromise between color-plot and isolines-plot is a color plot with stepwise color scheme:



It can be achieved by inactivating (# at the begin of the line) and using following commands:

```
#set contour base
set palette maxcolor 5 #color steps - alternative to the isolines. Use together with...
set cbrange [1:1.5] #fixed color scale (fixed range of the legend)
```

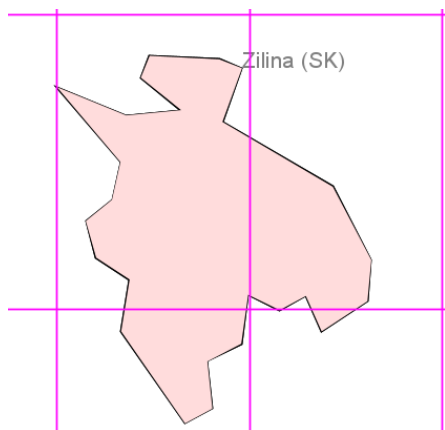
For more details see 'PlotFormatting.pdf' ('**Help/Plots formatting**' menu item).

Splitting/enlarging the city

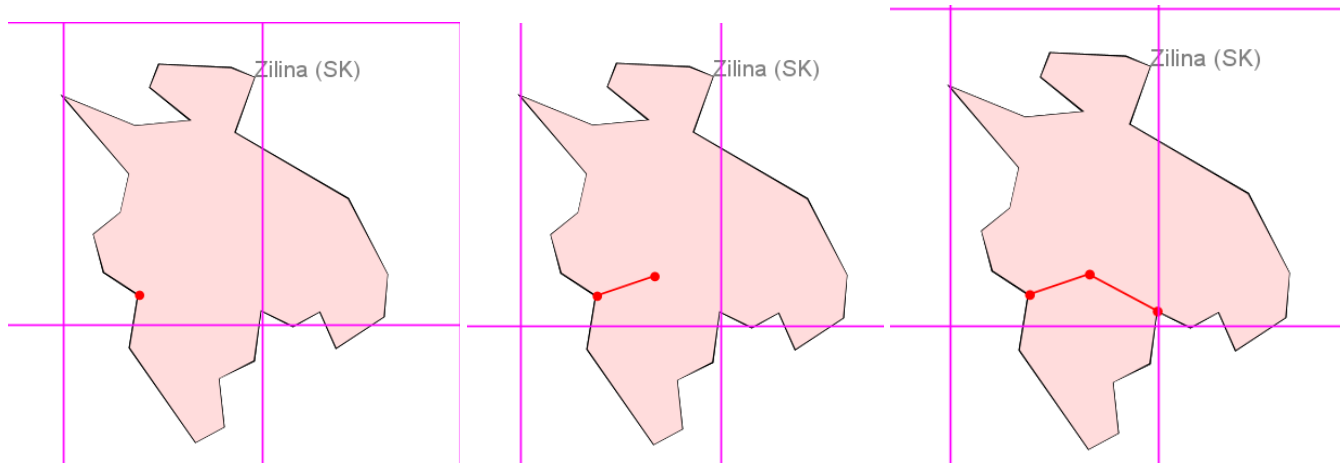
If you want to change parameters of any part of the city separately, we strongly recommend to prepare the parts of the city as separate city-files. However, some basic splitting/enlarging of the city is possible directly in the GUI. In any case, create a backup of the city-files before changing them in GUI.

How to split the city into two parts

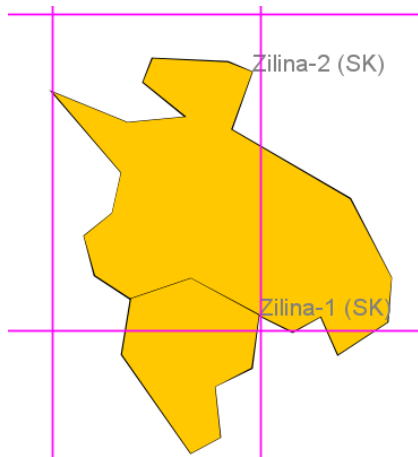
Select “**Tools/Enlarge/split a city (1 new vertex)**” menu item and a brief help will be displayed. Then **left-click** the city you want to split. The city changes the color to pink:



Then, select the first vertex of the cutting-line, e.g.:

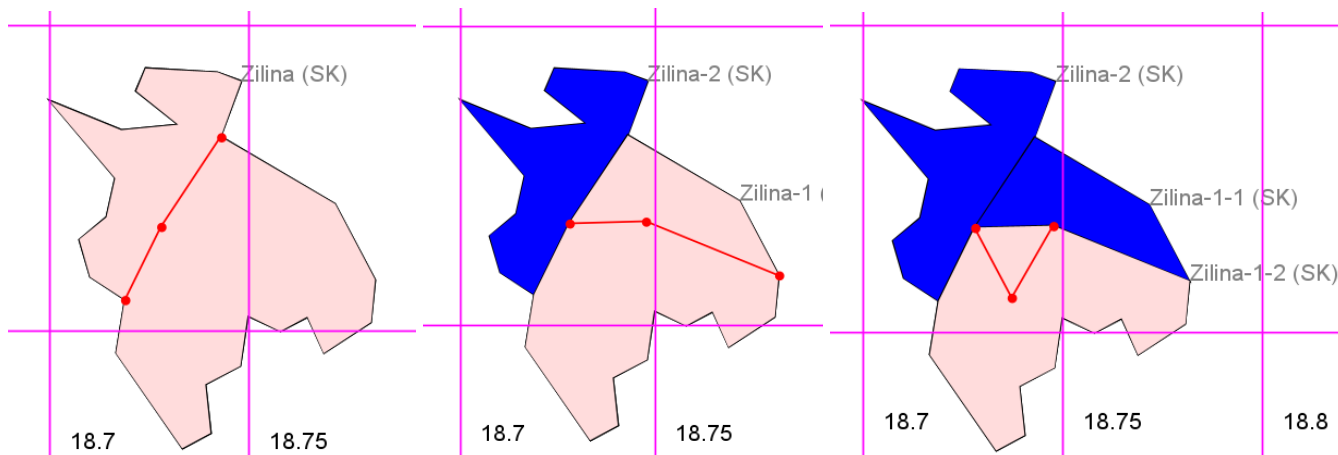


Then click any **point inside the city** and finally, another vertex of the city. In any step, right-click removes the last action. If you are happy with the cutting-line, double click the city:

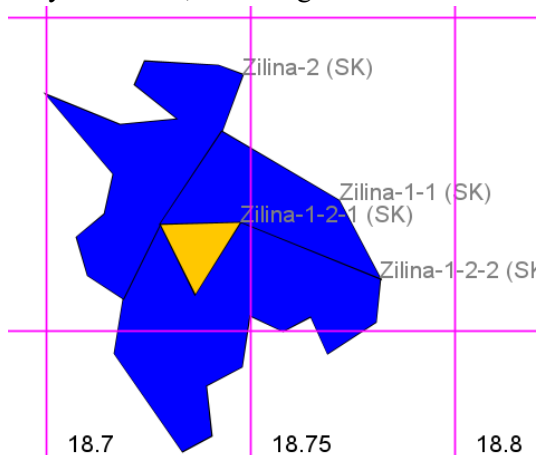


As you can see, the city “Zilina” is now split into 2 parts: “Zilina-1” and “Zilina-2”. The GUI automatically adjusted the total emission and/or the population of both parts according to their area. Now, the city emission function (total emission, angular dependence, spectrum) of any part can be set independently.

Although the GUI is only intended for simple slicing of the city, it is possible to carve out the inner parts of the city by repeating the splitting-process. As an example, we will separate a triangular area. The procedure is shown step by step in the next figure:



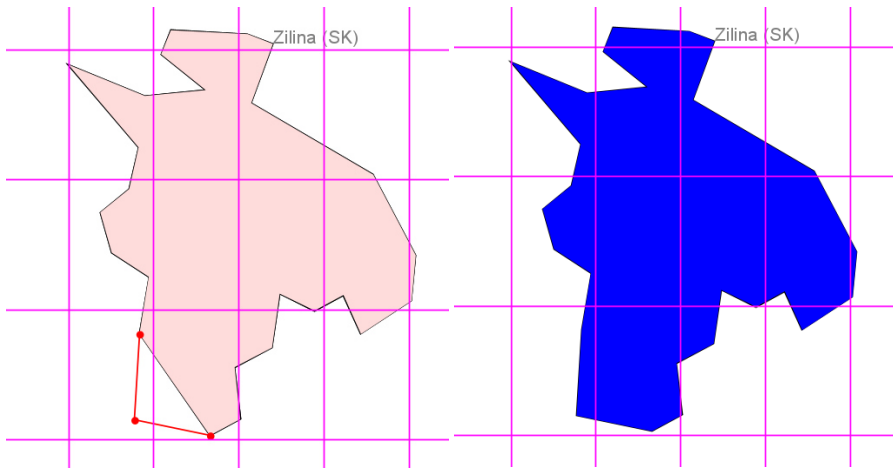
As you can see, the triangular area inside the city can be selected separately now:



Of course, analogically you can separate more complicated polygons.

How to enlarge the city

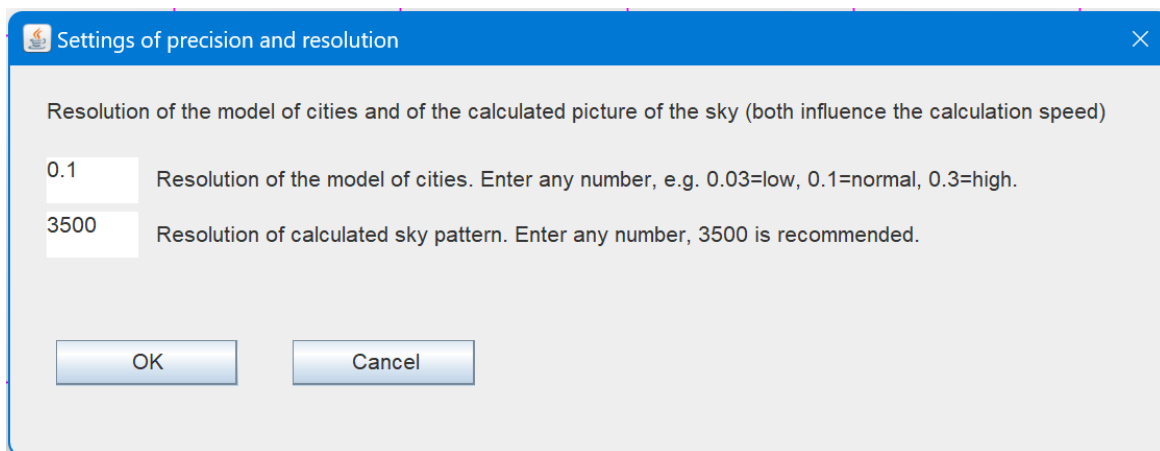
Select “**Enlarge/split a city (1 new vertex)**” radio button and **left-click** the city you want to split. The city changes the color to pink. Then, select the first vertex of the new border-line. Then click any **point outside the city** and finally, another vertex of the city. In any step, right-click removes the last action. If you are happy with the new border-line, double click the city:



The GUI automatically adjusted the total emission and/or the population according to the increased city area.

Setting the precision/resolution

Select the “Calculate/Set resolution...” menu item – a dialog will be displayed:



The first parameter defines the density of points sources placed inside the city polygons. The higher is the density, the smoother and more accurate is the calculated skyglow pattern. The point sources used in the calculation are shown in “city polygon *.png” file(s) stored in the “results” folder.

The second parameter represents the number of points in the hemisphere where the skyglow is calculated. This parameter controls the resolution of the calculated output only, it does not influence the precision of the calculated values.

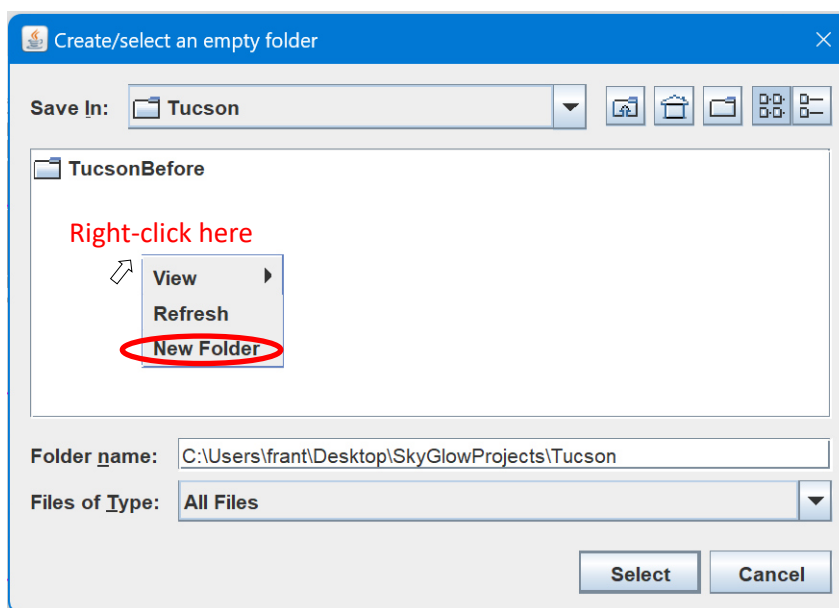
Both parameters significantly influence the calculation time. In the case the observer is outside of a city and far from biggest cities, try to decrease the density of the points in city-polygons to 0.01 or even lower value. Your calculations will be dramatically faster with no significant change in precision.

6. Archiving the projects

By default, data-files in subdirectories of the folder where program “SkyGlow.jar” is located, are used. The last saved settings are loaded automatically from “results/MSNsR_v6.cfg” file at the start of the “SkyGlow.jar” program. Eventually, use “File/Load last saved default project” menu item. Any changes in the project are stored into the “results/MSNsR_v6.cfg” file at the start of the solving the project or at using the “File/Save the parameters of current project” menu item. Don’t forget to save your settings regularly.

If you want to abandon current settings and results and to start a new project with the default parameters, use “File/New default project” menu item.

If you want to archive your model and results, use “File/Archive current project as...” menu item. You will be prompted to select an empty folder, where the project-files should be copied. Using right mouse button within the dialog window you can create a new empty folder:



After an empty folder was selected, all data-files (cities, spectra, aerosols, ...), configuration and results are copied into the selected folder. In the future, you can continue with work on your project, even if the default data-files are changed or removed. **It is recommended to use/create a folder outside of the SkyGlow installation folder (and outside the folder of currently opened project) to prevent recursively nested folders/subfolders from being created.**

To open an archived project, use “File/Load archived project...” menu item. All changes will be stored in the folder where the project is archived, and these changes do not affect the default project.

7. Creating own model

The present version of the program uses only few (test) models for aerosol particles, clouds, cities, ... We strongly encourage the users to create their own models and share it with whole community and send copies of new files to m.kocifaj@gmail.com. This is the best way how we can build the database for many regions worldwide. The file-structure we have built till now is self-explanatory. The data in *.cty.xml files are used for geographical latitude and longitude of all vertices that define the polygonal map of a city. At present, only two city emission functions (cefs) can be used: Garstang’s (includes surface reflectivity and fraction of light emitted

directly upwards) and cosine (zero direct uplight in Garstang's function). Data-file input is currently under development. Spectra of light sources can be defined in lights*.spc.xml files (the number of data is limited to 200). Spectral profiles of aerosol asymmetry parameter (*.asy.xml) and single scattering albedo (*.ssa.xml) can be found in directory "aerosols". The number of data is limited to 200. The same is true for spectral albedo of cloudy layer (see clouds*.cld.xml). The obstacles are simulated through horizon shielding – see the files horizons*.msk.xml where the <key> is the azimuth while the <value> is the elevation angle of an obstacle.

Note 1: All possible options are loaded during the start of the SkyGlow program. If you change the content of any configuration file (e.g. albedo of an aerosol) or create a new file (e.g. new spectral dependence of CEF), you have to restart the program.

Note 2: Setting the clear-sky or partly-cloudy conditions (many layers of the atmosphere have to be considered) and/or the photometric output (many wavelengths have to be calculated) can lead to more time-expensive calculations.

Note 3: At present, only one type of a city emission functions (CEF) can be used (Garstang). Data-input is under development.

Some hints how to prepare required data-files are collected below.

Creating the city-files

In the SkyGlow program, cities are modeled as surface light sources bounded by a polygon. A polygon is defined by the location (latitude and longitude) of its vertices. A simple way to obtain the bounding polygon of the city is to use the **Google Earth application**, which allows you to mark the so-called "path" and save it (including breakpoints) to a KML file in structured format. The KML file can be then converted into the XML format used by SkyGlow (city polygon vertices + metadata) in simple steps.

Note, that very detailed city boundary do not result in more accurate sky pattern. A few tenths of points is more than enough.

City digitization procedure:

1. Start the Google Earth program and zoom in on the city (for example, Malacky).
2. Select "Path" in the "Add"-menu, enter its name (for example "Malacky").
3. Use the mouse to mark the edge points of the city (a line is drawn with the vertices highlighted). The path need not to be closed, it will be closed automatically in later steps.
4. Click "OK" when finished.
5. Save the marked path (place) in KML format (for example as Malacky.kml) – don't use compressed KMZ format!

Comment: In **Google Earth WEB interface**, the process is very similar: create a polygon and save it locally. Then export the polygon as a KLM file. Next steps (conversion) are identical.

6. Open Tools/Convert KML-file into city-file menu item:

Conversion of KML-path to a city-file

Conversion of the KML path-file (Google Earth) to SkyGlow v.6 XML city-file

Malacky Full-name of the city (mandatory, will be shown in the map)

MA Shortcut of the city (optional)

Slovakia Full-name of the country (optional)

SK Shortcut of the country (optional, will be shown in the map if set)

18804 Population of the city (mandatory if 'Total lumens' is set to 0)

0 Total lumens produced by the city (0 if unknown)

HPS.spc.xml (High pressure sodium light source) Spectrum data file

0.15 Surface albedo (0 ... 1)

0.1 Fraction of light emitted directly to the upper hemisphere (0 ... 1)

Input KLM file: **Browse...** Malacky.kml

Output file name: Malacky.xml **Create**

Close KML converter

7. Fill-in all known information about the city. If 'total lumens' produced by the city is unknown, it will be estimated from the city population (typical value is 750 lm per habitant and this value can be changed later in the SkyGlow GUI). Typical value of surface albedo is 0.1-0.2. Cities with older types of luminaires have typical uplight 0.1-0.2. Modern LED luminaires have close-to-zero uplight. However, windows and monumental lighting produce uplight with typical value of 0.05-0.1 of total lumens produced by the city.
 8. Click "Browse" button and select the KML-file (e.g. 'Malacky.kml').
 9. Fill in the name of the city file (its extension must be .xml), e.g. 'Malacky.xml'.
 10. Press "Create" button. The city-file will be created and stored in 'cities'-folder.
 11. Repeat steps 6 – 10 for other KML-files.
 12. When finished, press "Close KML converter" button.
- Comment: To see newly created cities in SkyGlow application, close it and start again.

Creating the spectra of lights

The spectra of common light sources are available online or are distributed by the producers of light sources. The density of wavelength values in such spectra is usually not constant – higher density is near peaks. As SkyGlow calculates the luminance at wavelengths with 5 nm step, it is easy to skip some peaks. So, beside the change to XML format, the aim of the conversion is to create the spectrum with 5 nm step and with integrated values within the 5 nm intervals.

The spectrum is calculated from 350 nm to 800 nm to cover all visible spectrum. The values outside this interval are omitted. If the wavelength interval in source file is narrower, ends of the calculated spectrum are filled with zeros.

The conversion process is as follows:

1. Prepare spectrum in form of simple text file with wavelengths sorted in ascending order. Empty lines and lines starting with '#' are ignored. The values in a line must be separated by space or by tabulators (don't use comma separated values). Wavelengths can be given in nanometers, angstroms or micrometers. A comma or a dot can be used as decimal separator.

Example of an input file:

```
# 4000K LED spectrum
# wavelength[nm] value[-]
345 0.00190
350 0.00214
353 0.00232
355 0.00247
359 0.00289
...
```

Save this file as the text-file, "LED_4000K.txt" as an example.

2. Open 'Tools/Create spectrum-file from text-file' menu item:

Conversion of a text-file to spectrum-file

Conversion of the text-file containing spectrum ('wavelength' 'value' in separate lines) to SkyGlow v.6 XML spectrum-file.
Wavelengths has to be sorted from smallest to largest.
Lines starting with '#' and empty lines are ignored.
Spectrum from 350 nm to 800 nm with 5 nm step will be created.

Full-name of the spectrum (mandatory, will be shown in the SkyGlow application):
LED 4000K (Street lamp)

Any comment to the spectrum (optional):
Typical spectrum found on Internet

Input text-file: LED_4000K.txt

Wavelength in the text-file is given in

Output file name:

3. Fill-in the name of the spectrum and optionally any comment.
4. Browse for input text file (e.g. 'LED_4000K.txt').
5. Select in the combo-box the unit used in the input-file for wavelengths.
6. Fill-in the name of the output file (its extension must be '.xml'), e.g. 'LED_4000K.xml'.
7. Press 'Create' button.

Repeat steps 3-7 for other prepared text-files containing spectra. Press 'Close spectrum converter' when finished.
Comment: The newly created spectra are stored in 'lights'-folder. After restarting the SkyGlow application they will be selectable in dialogs.

Mixing the spectra of lights

The spectrum of a city is a mixture of spectra of individual light types (HPS, LED, incandescent, ...). If you know the amount of lumens produced by individual light types (or the percentage), you can create a mixed spectrum using 'Tools/Mix spectra into a new spectrum' menu item:

Mixing spectra to a new spectrum-file

Mixing spectra to a new spectrum-file according to amount of emitted lumens. The weight can be the amount of lumens emitted by the source or a number proportional to the amount of lumens (percentage, fraction, etc.).

Weights:

30	HPS.spc.xml (High pressure sodium light source)
60	LED.spc.xml (LED light source)
10	MetalHalide.spc.xml (Metal halide light source)
0.0	HPS.spc.xml (High pressure sodium light source)
0.0	HPS.spc.xml (High pressure sodium light source)
0.0	HPS.spc.xml (High pressure sodium light source)

Spectrum name: HPS 30% + LED 60% + Metal Halide 10% Mandatory, will be shown in the application

Output file name: HPS_30_LED_60_MH_10.xml

Description: Test spectrum

Create

Close spectrum mixer

1. Select the light sources and fill in corresponding amounts of lumens (or percentage of lumens).
2. Fill in the name of the spectrum, the filename of the new spectrum-file and (optionally) the description of the new spectrum.
3. Press 'Create' button.

Repeat previous steps if another mixed spectrum is needed. Press 'Close spectrum mixer' button to finish.

Comment: The newly created spectrum will be available in the SkyGlow application after its restart.

Creating the spectral reflectivity of clouds

The cloud '.xml' files are self-explanatory. <key> contains the wavelength and <value> contains the reflectivity.

Creating the horizon profiles

The horizon '.xml' files are self-explanatory. <key> contains the azimuth and <value> contains the elevation.

Creating the aerosol SSA and ASY files

The SSA '.xml' and ASY '.xml' files are self-explanatory. <key> contains the wavelength and <value> contains the SSA/ASY value. The typical spectral dependencies of SSA/ASY for aerosol at given location can be found in data from AERONET-stations. See link: https://aeronet.gsfc.nasa.gov/cgi-bin/draw_map_display_inv_v3
Note that the inverted data are available later (at least 1 year) after the re-calibration of the devices. Besides SSA and ASY data you can find here Aerosol Optical Depth (Extinction) at 500 nm needed by Skyglow software and from the spectral dependence of the AOD you can also obtain the Angstrom exponent.

8. Calibrating the city emission function using VIIRS-DNB data

Besides the surface reflectivity and fraction of light radiated directly upwards (see chapter 5), the total amount of lumens radiated by the city is the most important parameter determining the night-sky brightness. Although it can be estimated from total number of citizens, it can be determined also from night image of the city taken by VIIRS-DNB sensor. The satellite image represents the amount of light (in nW/cm²/sr) radiated towards the zenith from every single pixel. Knowing the surface reflectivity of the city and the fraction of light radiated directly to zenith, the total amount of lumens can be found from the Garstang function. Of course, the correction based on the relative spectral responses of the VIIRS-DNB sensor and of the human eye has to be applied. As a result, the calculated total amount of lumens depends also on the spectrum of the light emitted by the city.

So, before the calibration of the city emission function, following parameters must be set (see chapter 5):

- Spectral dependence of the emitted light
- Surface reflectivity (fraction of the reflected light)
- Direct uplight (fraction of the light radiated directly upwards)

For tests, the VIIRS data for region of Zilina-city is distributed with SkyGlow. So, you can continue with subchapter 'Calibrate the total lumens of cities'. You can also ask authors of the SkyGlow software for prepared 2023 VIIRS composite of whole world. Its zipped version has about 500 MB, so we will share a download link.

Download of the VIIRS data file

To eliminate the effect of clouds and Moon, composite images of whole world are best suitable for our calibration. However, these images are huge and big part of them are noisy dark pixels of oceans. Good option is to use median-masked composites, where pixels having values less than the noise level are set to zero. This reduces dramatically the size of the data file.

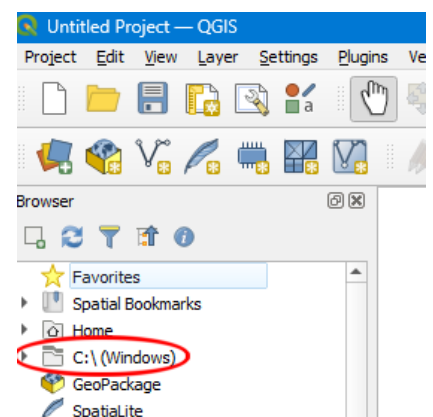
Such file (geotiff v2 „Median Masked File“) for year 2023 can be downloaded from the site https://eogdata.mines.edu/nighttime_light/annual/v22/2023/.

Files for other years are available, too. Beware, the size of the GeoTiff file can be more than 10 GB!

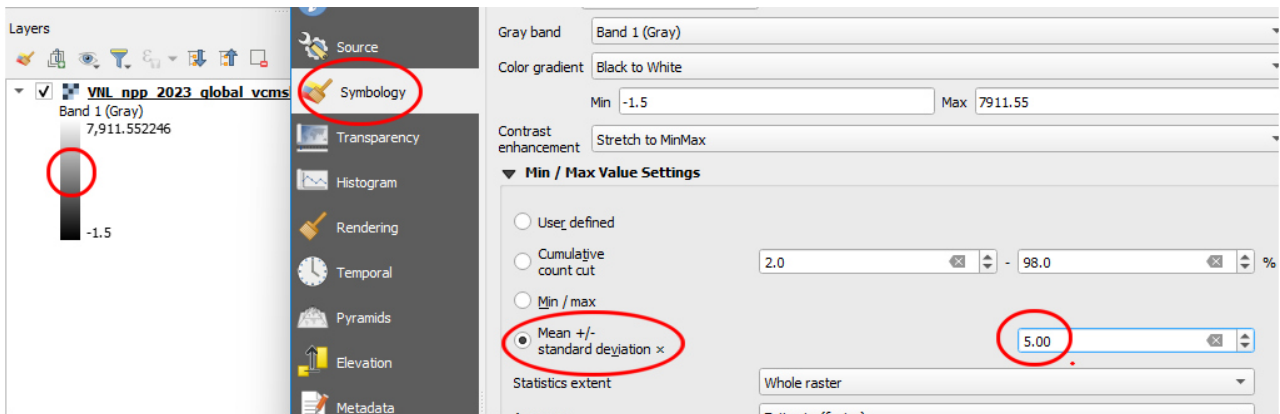
Data extraction in QDIS

The downloaded GeoTiff file can be processed in open QDIS-program. It is not necessary to work with huge data covering the entire world (about 1-2 GB in size), although that is also possible, because searching in SkyGlow is fast and the entire world is searched in about 1 minute. So, the first step is usually the extraction of the region where your cities are located.

1. Run the QDIS desktop program and create new project.
2. In the 'Browser'-window search for the downloaded file:

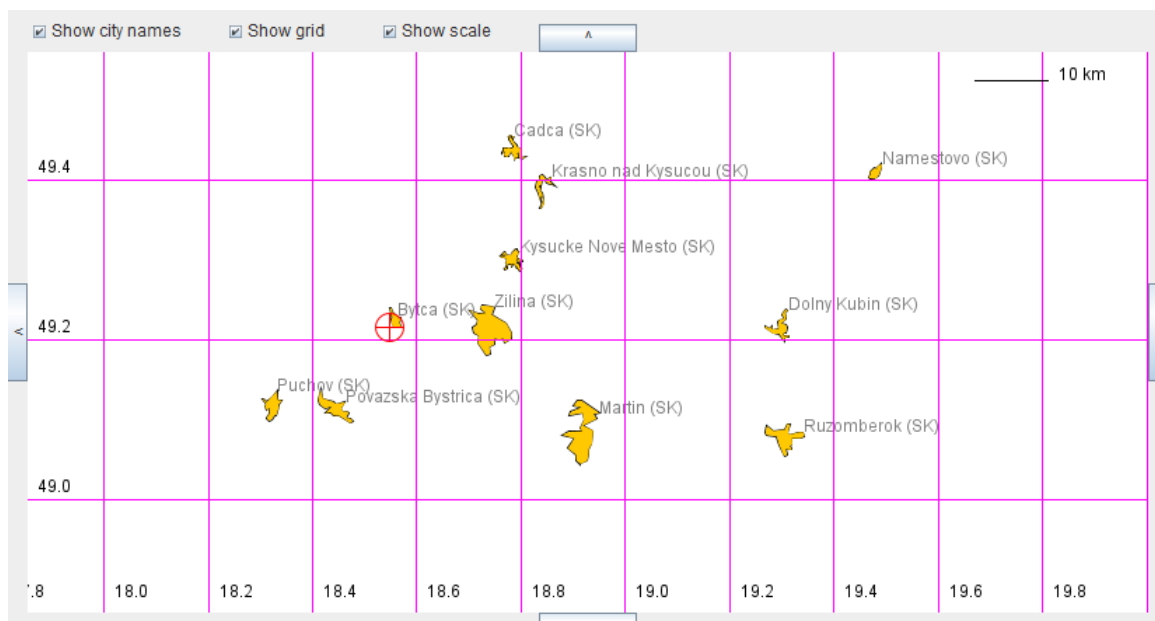


- For better view of the night data, double click the scale in 'Layers'-window, the select 'Symbology'-tab and set the Min/max as follows. Save the project (e.g. with name 'Whole_world') for future use.

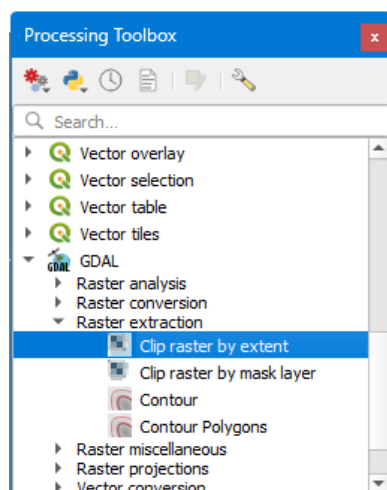


If you want to work with whole-world data, you can skip next steps and continue directly with step 2 in 'Export GeoTiff data into a standard text-file' section.

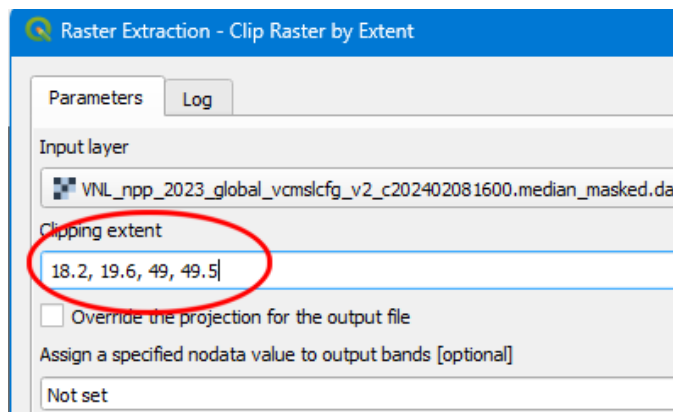
- Now we can extract our region. Look in the map in SkyGlow and identify the ranges of longitude/latitude where the cities are located, (18.2, 19.6) and (49.0, 49.5) in our case: .



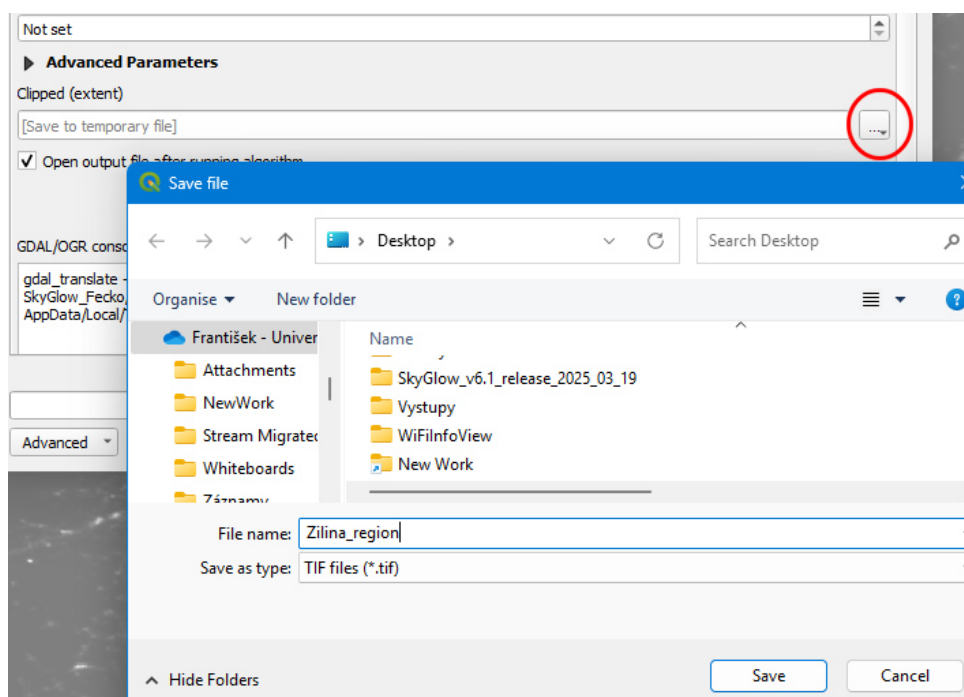
- In the 'Processing Toolbox' of the QDIS program select the GDAL-tool 'Clip raster by extent':



6. Fill-in coma-separated values found in step 5:



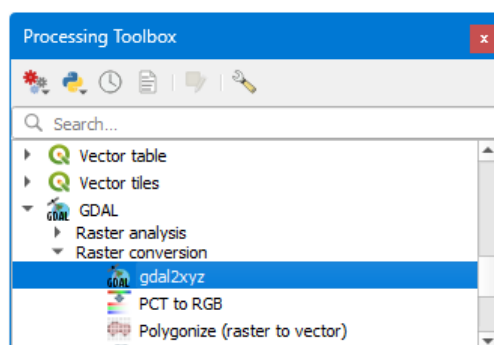
7. In the 'Advanced Parameters'-section click on the icon and select "Save to File...". Then choose the name of the new GeoTiff file (e.g. 'Zilina_region') and its location:



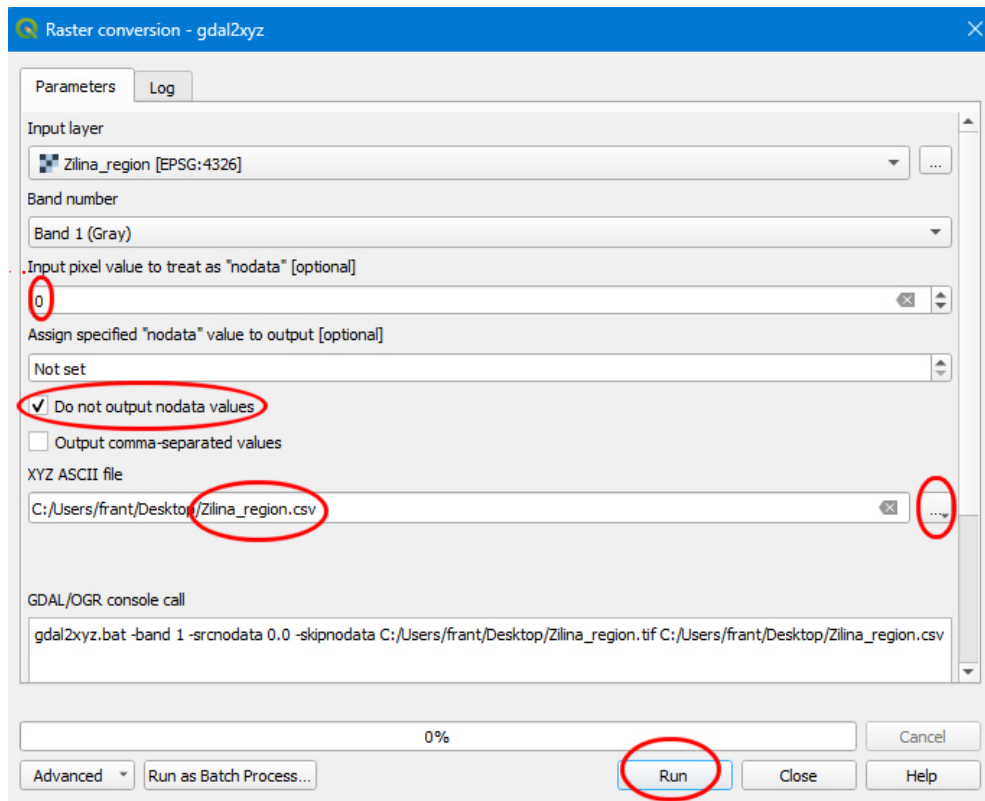
8. Finally, press the 'Run'-button to create new GeoTiff-file of selected region.

Export GeoTiff data into a standard text-file

1. Create a new QDIS project from newly created GeoTiff file of selected region (see steps 1-3 in previous subchapter).
2. In the 'Processing Toolbar' of the QDIS program select the GDAL-tool 'Raster conversion/gdal2xyz':



- Set the 'nodata'-value to 0 and check the 'Do not output nodata values'-checkbox. This will dramatically reduce the size of the output file. Choose the location and name (e.g. 'Zilina_region') of the output file (its extension will be automatically set to 'csv' and the format will be a standard space-separated text-file). Press the 'Run'-button. **Warning!** The processing time dramatically depends on the size of region to be exported. It can be a few seconds (as in our example) or many hours for whole world.

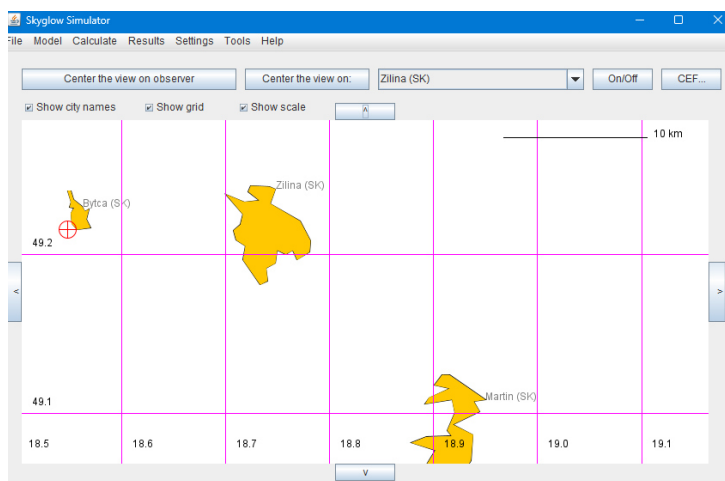


- Copy the newly created CSV file into the viirs-subfolder of the SkyGlow program.

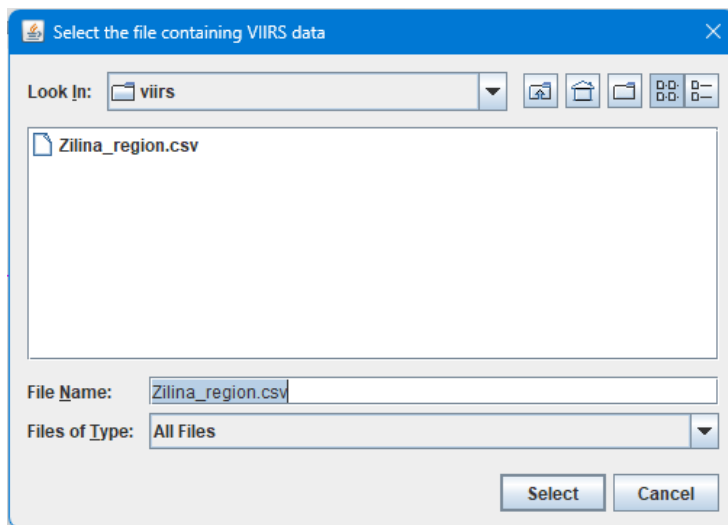
Calibrate the total lumens of cities

The final step is the calibration in SkyGlow program. It is not necessary to calibrate all cities. Only cities currently visible in the Skyglow-map will be processed. In addition, you will be asked to confirm the update of every single city (or you can choose automatic update of all cities currently shown in map). As an example, we will find the total lumen output of cities Zilina and Martin.

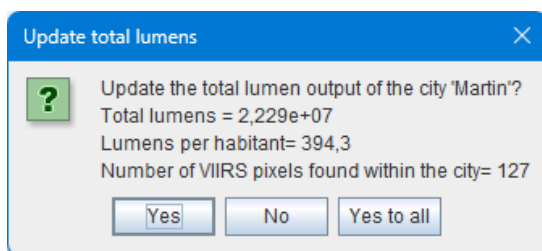
- Zoom-in the map until Zilina and Martin are at least partly visible. Ignore the 'Bytca'-city, we will skip this city later.



2. Select 'Tools/Calibrate total lumens of cities using VIIRS...' menu item. You will be warned about planned action, click 'Yes'.
3. In the File-Open dialog select the 'Zilina_region.csv' file created in previous subchapter.

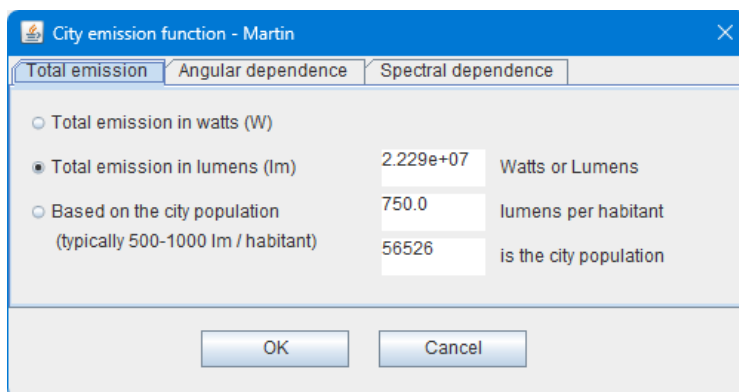


4. After a while the VIIRS-file is processed and you will be prompted to confirm update of individual cities. Firstly, press 'No' for Bytca-city. It was visible in map, but we do not want to update this city. Then, information about city of Martin is displayed:



We can see, that total amount of lumens was found to be 22.29 million. This value corresponds to 394 lumens per citizen (this value is typically a few hundred lumens per citizen). The city covers 127 VIIRS-pixels, so the statistics of the data is excellent.

5. Press 'Yes'-button and do the same for the city of Zilina.
6. Afterwards, you can right-click the 'Martin'-city on the map and check that the total lumen output was set as known and the calculated value was entered:



You can do the calibration again in the future or you can return to the estimate based on the city population (select corresponding radio-button and the total emission will be recalculated). To apply such return for a group of cities, see instructions in Chapter 5 at the end of Section "Changing the city emission function".

9. Troubleshooting the problems

There are two log-files generated by the computational modules: "logfile.txt" and "results/MSNsR_v5.log".

The first one stores the SkyGlow GUI outputs, such as information on your settings and actions performed before solver (or viewer) was launched. Error-messages are also stored here. Look into 'logfile.txt', if program crashes or some city is not displayed on the map and so on.

The second one contains logs from core computations.

Both log-files can be useful in searching the source of eventual problems. The core-logs are now heavily shortened to save the disk space. The full log is only activated in coding phase and should help developers in bug-fixing.