A simple two-dimensional parameterisation for Flux Footprint Prediction (FFP)

For details of the derivation of the footprint parameterisation, see
Kljun, N., P. Calanca, M.W. Rotach, H.P. Schmid, 2015: A simple two-dimensional parameterisation for Flux Footprint Prediction (FFP). Geosci. Model Dev., 8, 3695-3713. doi:10.5194/gmd-8-3695-2015.

Please acknowledge the source of your footprint estimates by citing the above article. Thanks!

How to use FFP matlab code

The FFP functions listed below are not meant to be stand-alone functions, but functions that can be called from within your own data processing code. For example, FFP can be called from within a loop of your own matlab function to calculate a series of footprints for a selected time series of your flux data.

1) To calculate a single FFP flux footprint use *calc_footprint_FFP*.

calc_footprint_FFP.m
[x_ci_max,x_ci,f_ci,x_2d,y_2d,f_2d,flag_err] = calc_footprint_FFP(zm,z0,umean,h,ol,sigmav,ustar)
FFP Input

zm	= Measurement height above displacement height (i.e. z-d) [m]
z0	= Roughness length [m] - enter [NaN] if not known
umean	= Mean wind speed at zm [ms-1] - enter [NaN] if not known
h	= Boundary layer height [m]
ol	= Obukhov length [m]
sigmav	= standard deviation of lateral velocity fluctuations [ms^{-1}]
ustar	= friction velocity [ms^{-1}]

Note: Either z0 or umean need to be entered. If both are given, umean is selected to calculate the footprint.

FFP output

x_ci_max	= x location of footprint peak (distance from measurement) [m]
x_ci	= x values of crosswind integrated footprint [m]
f_ci	= footprint function values of crosswind integrated footprint [m^{-1}]
x_2d	= x-grid of 2-dimensional footprint [m]
y_2d	= y-grid of 2-dimensional footprint [m]
f_2d	= footprint function values of 2-dimensional footprint [m^{-2}]
flag_err	= 1 in case of error, 0 otherwise

example: [x_ci_max,x_ci,f_ci,x_2d,y_2d,f_2d,flag_err] = calc_footprint_FFP(20,0.1,2000,-100,0.6,0.4)

calc_footprint_FFP_umean.m
[x_ci_max,x_ci,f_ci,x_2d,y_2d,f_2d,flag_err]=calc_footprint_FFP_umean(zm,umean,h,ol,sigmav,ustar)
FFP Input

zm	= Measurement height above displacement height (i.e. z-d) [m]
umean	= Mean wind speed at measurement height zm [ms^{-1}]
h	= Boundary layer height [m]
ol	= Obukhov length [m]

sigmav = standard deviation of lateral velocity fluctuations [ms^{-1}]
ustar = friction velocity [ms^{-1}]

FFP output

x_ci_max = x location of footprint peak (distance from measurement) [m]
x_ci = x values of crosswind integrated footprint [m]
f_ci = footprint function values of crosswind integrated footprint [m^{-1}]
x_2d = x-grid of 2-dimensional footprint [m]
y_2d = y-grid of 2-dimensional footprint [m]
f_2d = footprint function values of 2-dimensional footprint [m^{-2}]
flag_err = 1 in case of error, 0 otherwise

example: [x_ci_max,x_ci,f_ci,x_2d,y_2d,f_2d,flag_err] = calc_footprint_FFP_mean(20,3.5,2000,-100,0.6,0.4)

2) To derive the source area of R% of the flux footprint, call *calc_footprint_FFP_percentage* with the output of (1). This function can be used for a single value of R (e.g., 80%), or for an array of Rs (e.g., [20, 40, 60,80]%).

calc_footprint_FFP_percentage.m

[FFP,flag_err] = calc_footprint_FFP_percentage(x_2d,y_2d,f_2d,r)

Input for FFP percentage

x_2d,y_2d,f_2d = output of calc_footprint_FFP.m or calc_footprint_FFP_umean.m
r = percentage of footprint, i.e. a value between 10 and 90.
Can be either a single value (e.g., <80>) or an array of increasing percentage values (e.g., <[10:10:80]>)

Output of FFP percentage

FFP = structure array with footprint contour lines
FFP.r = percentage of footprint as in input
FFP.f = footprint value at r
FFP.x = x-array for contour line of r
FFP.y = y-array for contour line of r
For array of percentage values, structure entries can be accessed as FFP(1).r, FFP(1).x
flag_err = 1 in case of error, 0 otherwise

example: [FFP,flag_err] = calc_footprint_FFP_percentage(x_2d,y_2d,f_2d,[20:20:80])

3) To rotate a single flux footprint into the main wind direction, call *calc_footprint_FFP_rotated* with the output of (1).

calc_footprint_FFP_rotated.m

[x_2d_rot,y_2d_rot,f_2d_rot,flag_err] = calc_footprint_FFP_rotated(x_2d,y_2d,f_2d,wind_dir)

Input for FFP rotated

x_2d,y_2d,f_2d = FFP output of calc_footprint_FFP.m or calc_footprint_FFP_umean.m
wind_dir = wind direction in degrees (of 360)

Output of FFP rotated

x_2d_rot = rotated x-grid of 2-dimensional footprint [m]
y_2d_rot = rotated y-grid of 2-dimensional footprint [m]
f_2d_rot = rotated footprint function values of 2-dimensional footprint [m^{-2}]
flag_err = 1 in case of error, 0 otherwise

example: [x_2d_rot,y_2d_rot,f_2d_rot,flag_err] = calc_footprint_FFP_rotated(x_2d,y_2d,f_2d,120)

4) To rotate a single flux footprint into the main wind direction and to derive the source area of R% of this rotated flux footprint, call *calc_footprint_FFP_percentage_rotated* with the output of (1).

calc_footprint_FFP_percentage_rotated.m

[FFP_rot,flag_err] = calc_footprint_FFP_percentage_rotated(x_2d,y_2d,f_2d,r,wind_dir)

Input for FFP percentage rotated

x_2d,y_2d,f_2d	= output of calc_footprint_FFP.m or calc_footprint_FFP_umean.m
r	= percentage of footprint, i.e. a value between 10 and 90. Can be either a single value (e.g., "80") or an array of increasing percentage values (e.g., "[10:10:80]")
wind_dir	= wind direction in degrees (of 360)

Output of FFP percentage rotated

FFP_rot	= structure array with footprint contour lines
FFP_rot.x_2d	= rotated x-grid of 2-dimensional footprint [m]
FFP_rot.y_2d	= rotated y-grid of 2-dimensional footprint [m]
FFP_rot.f_2d	= rotated footprint function values of 2-dimensional footprint [m-2]
FFP_rot.r	= percentage of footprint as in input
FFP_rot.f	= footprint value at each r
FFP_rot.x	= x-array for contour line of each r
FFP_rot.y	= y-array for contour line of each r For array of percentage values, structure entries can be accessed as FFP_rot(1).r, FFP_rot(1).x
flag_err	= 1 in case of error, 0 otherwise

example: [FFP_rot,flag_err] = calc_footprint_FFP_percentage_rotated(x_2d,y_2d,f_2d,80,120)

5) For an aggregated footprint, i.e. a so-called footprint climatology, use either *calc_footprint_FFP* or *calc_footprint_FFP_umean* to derive footprints for each single time step, turn the footprints into the main wind direction per time step with *calc_footprint_FFP_rotated,* and aggregate the footprint-output raster f_2d_rot over a selected period of time (e.g. for specific months, seasons, or years). Finally, use *calc_footprint_FFP_percentage* to derive the respective contour lines for a source area of R%.

6) To plot the footprint in matlab, type, for example

crosswind-integrated footprint

```
plot(x_ci,f_ci,'k-',[x_ci_max x_ci_max],[0 max(f_ci)],'b--')
```

two-dimensional footprint with contour line of R%

```
surf(x_2d,y_2d,f_2d);shading flat;view(2);hold all
z=max(f_2d(:)).*1.2.*ones(size(FFP(1).y));
plot3(FFP(1).x,FFP(1).y,z,'r')
```

three-dimensional footprint surface

```
surf(x_2d,y_2d,f_2d);shading flat
```