# 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 R code

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

**1) Single footprint**

To calculate a single FFP flux footprint, call **calc_footprint_FFP** as described below. To rotate a single flux footprint into the main wind direction, call calc_footprint_FFP with an optional input value for the wind direction. To derive the source area of R% of the flux footprint, call calc_footprint_FFP with an optional additional single value of R (e.g., 80 or 0.8 for 80%), or with an array of Rs (e.g., c(20, 40, 60, 80) or seq(10, 80, 10)).

FFP <- calc_footprint_FFP(zm,z0,umean,h,ol,sigmav,ustar)

**FFP Input**

All inputs as scalars

| | |
|---|---|
| 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 is required. If both are given, z0 is selected to calculate the footprint.

*Optional input:*

| | |
|---|---|
| wind_dir | = Wind direction in degrees (of 360) for rotation of the footprint |
| r | = Percentage of source area, i.e. a value between 10% and 90%.<br>Can be either a single value (e.g., "80") or an array of increasing percentage values (e.g., " seq(10, 80, 10)")<br>Expressed either in percentages ("80") or in fractions of 1 ("0.8")<br>Default is [10:10:80]. Set to "NaN" for no output of percentages |
| nx | = Integer scalar defining the number of grid elements of the scaled footprint. Large nx results in higher spatial resolution and higher computing time. Default is 1000, nx must be >=600. |
| rslayer | = Calculate footprint even if zm within roughness sublayer: set rslayer = 1. Note that this only gives a rough estimate of the footprint as the model is not valid within the roughness sublayer. Default is 0 (i.e. no footprint for within RS). z0 is needed for estimation of the RS. |
| crop | = Crop output area to size of the 80% footprint or the largest r given if crop=1 |

**FFP output**

| | |
|---|---|
| FFP | = Structure array with footprint data with footprint data for measurement at [0 0 zm] m |
| x_ci_max | = x location of footprint peak (distance from measurement) [m] |
| x_ci | = x array 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], rotated if wind_dir is provided |
| y_2d | = y-grid of 2-dimensional footprint [m], rotated if wind_dir is provided |
| f_2d | = footprint function values of 2-dimensional footprint [$m^{-2}$] |
| r | = percentage of footprint as in input, if provided |
| fr | = footprint value at r, if r is provided |
| xr | = x-array for contour line of r, if r is provided |
| yr | = y-array for contour line of r, if r is provided |
| flag_err | = 1 in case of error, 0 otherwise |

**Example**

```
FFP <- calc_footprint_FFP(zm=20,z0=0.01,h=2000,ol=-100,sigmav=0.6,ustar=0.4, wind_dir=30,
                          r=seq(10,80,10))
```

**2) Single footprint within a given, fixed domain**

In some cases it may be useful to derive a footprint for a pre-set given domain. For such a case, use **calc_footprint_FFP_climatology** with a single set of input parameters. For details of input and output parameters, see Section 3 below.

**3) Footprint climatology**

A footprint climatology is an aggregation of footprints over several time steps. To calculate a footprint climatology with FFP, call **calc_footprint_FFP_climatology** as described below. Again, optional input parameters can be provided to, for example, derive the source area of R% of the flux footprint climatology, call calc_footprint_FFP_climatology with an optional additional single value of R (e.g., 80 for 80%), or with an array of Rs (e.g., c(20, 40, 60, 80) or seq(10, 80, 10)). You can also plot an example figure of your footprint climatology by setting fig = 1.
This function calculates footprints within a fixed physical domain (either default area or user input). For determining the optimal extent of the domain (large enough to include the footprints) use the function calc_footprint_FFP as described in Section 1.

Important: to run calc_footprint_FFP_climatology, the "spatialfil" and "EBImages" packages need to be installed. See Section 5 for information on how to install EBImages.

**FFP Input**

All vectors need to be of equal length (one value for each time step, scalars possible)

| | |
|---|---|
| zm | = Measurement height above displacement height (i.e. z-d) [m]<br>Usually a scalar, but can also be a vector |
| z0 | = Roughness length [m] - enter [NaN] if not known<br>Usually a scalar, but can also be a vector |
| umean | = Mean wind speed at zm [ms-1] - enter [NaN] if not known<br>Either z0 or umean is required. If both are given, z0 is selected to calculate the footprint |
| h | = Vector of boundary layer height [m] |

ol = Vector of Obukhov length [m]
sigmav = Vector of standard deviation of lateral velocity fluctuations [$ms^{-1}$]
ustar = Vector of friction velocity [$ms^{-1}$]
wind_dir = Vector of wind direction in degrees (of 360) for rotation of the footprint

*Optional input:*
domain = Domain size as an array of (xmin xmax ymin ymax) [m].
Footprint will be calculated for a measurement at [0 0 zm] m
Default is smallest area including the r% footprint or (-1000, 1000, -1000, 1000) m, whichever smallest (80% footprint if r not given).
dx, dy = Cell size of domain [m]
Small dx,dy result in higher spatial resolution and higher computing time
Default is dx = dy = 2 m (if neither domain nor nx and ny are given).
If only dx is given, dx=dy.
nx, ny = Two integer scalars defining the number of grid elements in x and y
Large nx and ny result in higher spatial resolution and higher computing time
Default is nx = ny = 1000. If only nx is given, nx=ny
If dx,dy and nx,ny are given, dx,dy is given priority
r = Percentage of source area, 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., " seq(10, 80, 10)")
Expressed either in percentages ("80") or in fractions of 1 ("0.8")
Default is [10:10:80]. Set to "NaN" for no output of percentages
rslayer = Calculate footprint even if zm within roughness sublayer: set rslayer = 1. Note that this only gives a rough estimate of the footprint as the model is not valid within the roughness sublayer. Default is 0 (i.e. no footprint for within RS). z0 is needed for estimation of the RS.
smooth_data = Apply convolution filter to smooth footprint climatology if smooth_data=1 (default)
crop = Crop output area to size of the 80% footprint or the largest r given if crop=1
pulse = Display progress of footprint calculations every pulse-th footprint (e.g., "100")
fig = Plot an example figure of the resulting footprint (on the screen): set fig = 1.
Default is 0 (i.e. no figure)

**FFP output**
FFP = Structure array with footprint data with footprint data for measurement at [0 0 zm] m
x_2d = x-grid of footprint climatology [m]
y_2d = y-grid of footprint climatology [m]
fclim_2d = Normalised footprint function values of footprint climatology [$m^{-2}$]
r = Percentage of footprint as in input, if provided
fr = footprint value at r, if r is provided
xr = x-array for contour line of r, if r is provided
yr = y-array for contour line of r, if r is provided
n = Number of footprints calculated and included in footprint climatology
flag_err = 1 in case of error, 2 if not all contour plots (r%) within specified domain, 0 otherwise
If the source area is calculated for 20%, 40%, 60% and 80%, and the 80% contour is extending further than the domain (but the other r's are within the domain), flag_err = 2 and all results are provided apart from those for the contour at 80%.

**Example**
```
FFP <- calc_footprint_FFP_climatology(zm=20, z0=0.01, umean=NA, h=c(2000,1800,1500),
                ol=c(-10,-100,-500), sigmav=c(0.9,0.7,0.3), ustar=c(0.5,0.3,0.4), wind_dir=c(30,50,70),
                domain=c(-100,1000,-100,1000), nx=1100, r=seq(10,80,10), smooth_data=1)
```

**4) Plotting footprints**

To plot the footprint climatology, you can set fig=1 when calling it. Or, in R, type, for example

Crosswind-integrated footprint

```
plot(FFP$x_ci,FFP$f_ci, type="l")
```

Two-dimensional view of single footprint (from calc_footprint_FFP.R) with contour lines of R% (using the fields package). Note that nx and ny of quilt.plot need to be adjusted for your output.

```
ffp_x <- c(FFP$x_2d)
ffp_y <- c(FFP$y_2d)
ffp_f <- c(FFP$f_2d)
quilt.plot(ffp_x,ffp_y,ffp_f,nx=1000,ny=1000, xlim=c(-100,1000),ylim=c(-100,1000))
for (i in 1:8) lines(FFP$xr[[i]],FFP$yr[[i]], type="l", col="red")
```

Two-dimensional view of footprint climatology with contour lines of R%.

```
image.plot(FFP$x_2d[1,], FFP$y_2d[,1], FFP$fclim_2d)
for (i in 1:8) lines(FFP$xr[[i]], FFP$yr[[i]], type="l", col="red")
```

Three-dimensional footprint climatology surface (using the plot3D package)

```
surf3D(FFP$x_2d, FFP$y_2d,FFP$fclim_2d)
```

Please note that the plotting convention for matrices varies with software package or even with the selected plotting command, i.e. point (1/1) of the matrix may be the lower left corner or the upper left corner. It hence is suggested that **the footprint plot is always checked against a wind rose**. For complex footprint climatologies, it is sufficient to check just one single footprint. It may be necessary to transpose the footprint matrix depending on the plotting tool.

**5) Installing EBImage and spatialfil**

For help, see https://github.com/aoles/EBImage/issues/2
In case of problems, please refer to R-specific help pages.

in R:

```
> source("https://bioconductor.org/biocLite.R")
> biocLite("EBImage")
> library(EBImage)
> install.packages("spatialfil")
```