The Journal of Open Source Software

prosail: an R package to simulate canopy reflectance with the coupled model PROSAIL (PROSPECT + SAIL)

Jean-Baptiste Féret [1¶] **and Florian de Boissieu** [1]

1 TETIS, INRAE, AgroParisTech, CIRAD, CNRS, Université Montpellier, Montpellier, France ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 31 January 2026
Published: unpublished

Summary

The PROSAIL model is a widely used radiative transfer model that combines the *leaf optical PROperties SPECtra* (PROSPECT) model with the *Scattering by Arbitrarily Inclined Leaves* (SAIL) canopy bidirectional reflectance model (Berger et al., 2018; Jacquemoud et al., 2009). It is designed to simulate the reflectance of vegetation canopies across a broad spectral domain, including the visible, near-infrared, shortwave infrared and thermal infrared regions of the electromagnetic spectrum.

PROSAIL integrates the optical properties of individual leaves, calculated by PROSPECT (Jacquemoud & Baret, 1990), with the structural characteristics of the canopy modeled by SAIL (W. Verhoef, 1984). This combination allows for the simulation of how light interacts with vegetation at both the leaf and canopy levels, taking into account factors such as leaf chemical content, leaf structure, canopy architecture, and viewing and illumination angles.

PROSAIL's ability to simulate plant canopy spectral and directional reflectance under various conditions makes it a valuable tool for agricultural monitoring, ecological studies, and climate research, where PROSAIL can be used to simulate vegetation optical properties (Poulter et al., 2023). It is also used in combination with inversion procedures to retrieve vegetation biophysical properties. These biophysical properties include leaf area index (LAI), leaf water and pigment content, the fraction of absorbed photosynthetically active radiation (fAPAR), albedo and fractional vegetation cover (fCover) among others. The capacity to retrieve a biophysical property mainly depends on the spectral characteristics of the sensor, and the acquisition of sufficient spectral information to solve an ill-posed problem.

Here, we introduce `prosail`, an R package which provides multiple versions of the model PROSAIL, coupled with the latest versions of PROSPECT distributed with the R package `prospect` (Féret & Boissieu, 2024). The package includes functions to run simulations of surface reflectance in forward mode for various optical sensors, as well as functions to apply user defined spectral response function (SRF), allowing simulation of any optical sensor with known or approximated surface reflectance. It also includes functions for model inversion using iterative optimization with and without prior information, as well as a module dedicated to hybrid inversion, combining physical modeling and machine learning regression, applicable to data tables and raster data.

Statement of need

The capacity to measure, map and monitor vegetation traits corresponding to biophysical and chemical properties is crucial to better understand ecosystem and agrosystem functions, as

well as carbon, water and energy budgets. Vegetation radiative transfer models (RTMs) aim at describing the interactions between light and these biophysical and chemical properties, through absorption and scattering mechanisms. At leaf scale, the model PROSPECT (Féret & Boissieu, 2024) is currently the most popular physical model for the simulation of leaf optical properties and the accurate estimation of leaf chemistry (Féret et al., 2019; Spafford et al., 2021). At canopy scale, various models can be employed, and the choice for a specific model may be driven by the complexity of the system to be simulated, as well as the capacity to describe the scene.

The SAIL model is an example of four-stream representations of the radiative transfer equation, including two direct fluxes (incident solar flux and radiance in the viewing direction) and two diffuse fluxes (upward and downward hemispherical flux). A system of four linear differential equations can then be analytically solved. Detailed description of the functioning of the different SAIL versions can be found in (Wout Verhoef & Bach, 2007) and (W. Verhoef et al., 2007).

More complex models include the *Soil Canopy Observation of Photosynthesis and Energy fluxes* (SCOPE) (Yang et al., 2021) model, which features layered description of the vegetation, enabling the simulation of vegetation with an understorey or with a vertical gradient in leaf properties, and the *Discrete Anisotropic Radiative Transfer* DART model (Gastellu-Etchegorry et al., 2015), which allows for explicit 3-dimensional description of the canopy using geometric representations with triangular meshes or turbid voxels that can be derived from LiDAR acquisitions.

RTMs are increasingly used in remote sensing applications, in combination with machine learning: RTMs are used to produce a surface reflectance dataset with corresponding vegetation properties, which are then used as training data to adjust a regression model with a machine learning algorithm. Such strategies, commonly referred to as hybrid methods (Verrelst et al., 2015), show multiple advantages : simulations are used in place of extensive in situ sampling, allowing regression model adjustment on virtually any source of optical imagery and any training sample size, while the use of machine learning ensures computation efficiency compared to iterative optimization when processing imagery data.

The relatively limited number of input parameters and the computational efficiency of the SAIL model makes it particularly interesting for operational applications, despite the main assumption of vegetation as a homogeneous turbid medium, where leaves are randomly distributed within the canopy, which is not accurate for row crops and heterogeneous canopies.

State of the field

Various softwares currently provide procedures for hybrid inversion with PROSAIL simulations. The *Sentinel Toolbox Application Platform* (SNAP) includes the `Biophysical Processor` module (Weiss et al., 2020), which provides an implementation of PROSAIL hybrid inversion combining PROSAIL simulations with an artificial neural network regression model, for the estimation of vegetation properties, such as LAI, fAPAR, fcover, canopy chlorophyll content and canopy water content. The sampling strategy for the definition of the training set, including the distribution and co-distribution of the input parameters, is described in the *Algorithmic Theoretical Background Document* (ATBD) (Weiss et al., 2020)) and cannot be adjusted by users. Alternative distributions provide more interactive parameterization of the inversion strategy, such as the *Automated Radiative Transfer Models Operator* (ARTMO) toolbox (Caicedo et al., 2014) developed in Matlab.

The R package `prosail` includes the versions of the model PROSPECT implemented in the package `prospect`: PROSPECT-D (Féret et al., 2017) and PROSPECT-PRO (Féret et al., 2021). It also includes two versions of the model SAIL:

- *4SAIL* (W. Verhoef et al., 2007), a numerically stable adaptation of the *SAILH* version

(W. Verhoef, 1998), which incorporate foliage hot spot to the original SAIL model

- *4SAIL2* (Wout Verhoef & Bach, 2007), a two layers version of *4SAIL* accommodating horizontal and vertical heterogeneities. This version includes additional parameters to account for crown clumping, and to describe the vertical distribution of two types of leaves described by specific leaf optical properties, which may describe different phenological stages or different vegetation types.

The R package `prosail` also provides a set of functions to help researchers aiming at experimenting for the estimation vegetation biophysical properties from optical remote sensing using RTM with open-source solutions. These functions include :

- the simulation of surface reflectance for any optical sensor based on their SRF
- the production of look-up-tables (LUTs) for simulated sensor surface reflectance and corresponding PROSAIL input parameters, with or without additive and multiplicative noise
- the application of iterative optimization with possible added prior information on data tables
- the application of hybrid inversion on data tables and raster data

The package `prosail` does not intend to provide the same computational efficiency as SNAP. It does not provide a collection of models and methods as comprehensive as those provided with the ARTMO box neither. `prosail` provides a flexible ope source framework to experiment with hybrid inversion procedures, using simple yet fast and efficient training stage, allowing experimenting on strategies for training data sampling, introduction of noise, feature selection over any type of optical sensor. It is particularly suitable for research and development, in order to identify potential and limitations of inversion strategies and parameterizations. Resulting regression models can be applied on remote sensing data, but it may not be the appropriate software for scaling up vegetation monitoring applications.

Alternative PROSAIL implementations can be found at this webpage. This includes distributions in matlab, python and fortran programming languages. Note that alternative PROSAIL implementations are also available in packages written in python, Julia and R.

Software design

`prosail` uses the R package `prospect` (Féret & Boissieu, 2024) for the simulation of leaf optical properties. It provides a user-friendly and modular set of functions, in order to run simulations of vegetation canopy reflectance, and to predict vegetation biophysical properties using various inversion strategies. Model inversion is a multi-step procedure requiring simulation of sensor reflectance in order to train a machine learning regression algorithm, then applicable to any new data source, including raster data from airborne and spaceborne sensors.

- The generation of realistic reflectance simulations requires definition of input parameter sampling strategy, accounting for sensor SRF and uncertainties. `prosail` provides functions to produce this training dataset as a unique step following standard parameterization, or step by step, in order for users to understand the logic, or adjust each step to their needs.
- A limited number of machine learning algorithms is currently provided, but the software design will allow integration of additional algorithms in future versions.
- Raster processing is mainly handled with the `terra` (Hijmans, 2025) package, to ensure compatibility with most raster data formats.
- `prosail` can be used in combination with `preprocS2`, a package dedicated to downloading and preprocessing of Earth observation data following the STAC specification, in order

to produce a fully automated image access and processing workflow.

Research impact statement

`prosail` has been used in multiple research publications since its early developments (Hauser et al., 2021), (Ferreira et al., 2026), (Féret et al., 2026). It is currently used in multiple research projects, and is actively maintained.

Overview

PROSAIL simulation in forward mode

PROSAIL requires information intrinsic to vegetation :

- leaf optical properties, including leaf directional hemispherical reflectance and transmittance. These leaf optical properties are simulated with the R package `prospect` (Féret & Boissieu, 2024) and readers are invited to refer to the documentation of this package for a comprehensive description of the leaf chemical and structure parameters accounted for by the various versions of the PROSPECT model. Measured leaf optical properties can also be used instead of simulated ones.
- LAI, which is defined as the one-sided green leaf area per unit ground surface area in broadleaf canopies. It is dimensionless and is sometimes expressed in $m^2.m^{-2}$
- Leaf inclination distribution, which can be defined following different *Leaf Inclination Distribution Functions* (LIDF). The LIDF used in the original version of SAIL [verhoef1998] describes leaf inclination based on two parameters: `lidf_a` controls the average leaf slope, and `lidf_b` controls the distribution's bimodality. Both parameters should be set between -1 and 1 and the sum of absolute values inferior to 1. An alternative LIDF defined by (Campbell, 1990) can be applied, and requires a unique parameter corresponding to the average leaf angle.
- a foliage hot spot parameter which corresponds to the ratio of the correlation length of leaf projections in the horizontal plane and the canopy height (Bréon et al., 2002; Kuusk, 1991; Wout Verhoef & Bach, 2007)

PROSAIL also requires information extrinsic to vegetation :

- the sun-observer geometry, defined by the sun zenith angle, the observer zenith angle, and the relative azimuth angle between sun and observer
- the soil reflectance

Additional parameters are required to run the version `4SAIL2` :

- the optical properties corresponding to the second type of leaves
- the fraction of LAI corresponding to a second type of leaves
- a layer dissociation factor between the main leaf layer and the second leaf layer
- the vertical crown cover percentage, which corresponds to the percentage of ground area covered with crowns as seen from nadir direction
- a tree shape factor corresponding to the ratio of crown diameter to crown height

SAIL produces a list of output variables corresponding to four top-of-canopy reflectance factors. Additional parameters required to compute albedo, fAPAR and fcover are also provided as outputs. The full list of outputs is then :

- `rddt`: bi-hemispherical reflectance factor

- `rsdt`: directional-hemispherical reflectance factor for solar incident flux
- `rdot`: hemispherical-directional reflectance factor in viewing direction
- `rsot`: bi-directional reflectance factor
- `abs_dir`: canopy absorptance for direct solar incident flux
- `abs_hem`: canopy absorptance for hemispherical diffuse incident flux
- `fcover`: Fraction of green Vegetation Cover (equals to 1 - beam transmittance in the target-view path)
- `rsdstar`: contribution of direct solar incident flux to albedo
- `rddstar`: contribution of hemispherical diffuse incident flux to albedo

The function `prosail` using `4SAIL` can be called as follows :

```
# Load prosail package
library(prosail)

# define PROSPECT input variables. Refer to prospect tutorial for default values
input_prospect <- data.frame('chl' = 40, 'car' = 8, 'ant' = 0.0,
                             'ewt' = 0.01, 'lma' = 0.009, 'n_struct' = 1.5)

# define input variables for SAIL.
lai <- 5        # LAI
hotspot <- 0.1  # foliage hot spot parameter
type_lidf <- 2  # leaf inclination distribution function : Campbell
lidf_a <- 30    # average leaf angle (degrees)
tts <- 30       # geometry of acquisition: sun zenith angle (degrees)
tto <- 10       # geometry of acquisition: observer zenith angle (degrees)
psi <- 90       # geometry of acquisition: sun-observer azimuth (degrees)
rsoil <- spec_soil_ossl$soil_01 # soil reflectance selected from OSSL library

# run PROSAIL with 4SAIL
ref_4sail <- prosail(input_prospect = input_prospect,
                     type_lidf = type_lidf, lidf_a = lidf_a, lai = lai,
                     hotspot = hotspot, tts = tts, tto = tto, psi = psi,
                     rsoil = rsoil)
```

Additional input variables are required when using `4SAIL2` :

```
# define leaf chemical content corresponding to two types of leaves
input_prospect <- data.frame('chl' = c(40,5), 'car' = c(8,5), 'ant' = c(0,1),
                             'ewt' = c(0.01,0.005), 'lma' = c(0.009,0.008),
                             'brown' = c(0.0,0.5), 'n_struct' = c(1.5,2))

# define additional 4SAIL2 parameters
fraction_brown <- 0.5   # fraction of LAI corresponding to second type of leaves
diss <- 0.5             # layer dissociation factor between the leaf layers
cv <- 1                 # vertical crown cover percentage
zeta <- 1               # tree shape factor

# run PROSAIL with 4SAIL2
ref_4sail2 <- prosail(SAILversion = '4SAIL2', input_prospect = input_prospect,
                      type_lidf = type_lidf, lidf_a = lidf_a, lai = lai,
                      hotspot = hotspot, tts = tts, tto = tto, psi = psi,
```

```
                                rsoil = rsoil, fraction_brown = fraction_brown,
                                diss = diss, cv = cv, zeta = zeta)
```

Computation of surface reflectance from PROSAIL outputs

The surface reflectance results from the combination of hemispherical-directional and bi-directional reflectance factors with the relative contribution of direct and diffuse radiation. The direct and diffuse radiations are defined as described by (C. François et al., 2002). Users can then compute surface reflectance by setting a global contribution of diffuse skylight radiation with the `skyl` parameter, which is the share of diffuse flux in global radiation. Alternatively, the direct and diffuse radiations can taken into account by computing `skyl` based on the equation proposed by (Spitters et al., 1986), knowing the sun zenith angle and assuming clear sky conditions.

```
# Compute surface reflectance with known skyl
surf_refl_4sail <- get_surf_refl(rdot = ref_4sail$rdot,
                                 rsot = ref_4sail$rsot,
                                 skyl = 0.23,
                                 spec_atm_sensor = spec_atm)


# Compute surface reflectance using sun zenith angle (clear sky conditions)
surf_refl_4sail <- get_surf_refl(rdot = ref_4sail$rdot,
                                 rsot = ref_4sail$rsot,
                                 tts = 40,
                                 spec_atm_sensor = spec_atm)
```

Computation of fAPAR and albedo

The fAPAR in the direction of the observer can be computed by combining both canopy absorptance for direct solar incident flux and canopy absorptance for hemispherical diffuse incident flux.

```
fapar_4sail <- get_fapar(abs_dir = ref_4sail$abs_dir,
                         abs_hem = ref_4sail$abs_hem,
                         tts = tts, spec_atm_sensor = spec_atm)
```

Finally, the albedo can be derived from direct solar incident flux and hemispherical diffuse incident flux.

```
albedo_4sail <- get_albedo(rsdstar = ref_4sail$rsdstar,
                           rddstar = ref_4sail$rddstar,
                           tts = tts, spec_atm_sensor = spec_atm)
```

Simulating surface reflectance acquired by a sensor

The simulation of sensor surface reflectance requires the SRF of the sensor. A selection of SRF corresponding to multiple sensors is already implemented in `prosail` :

- Sentinel-2A, 2B and 2C
- Landsat-7, 8 and 9
- MODIS
- Venus
- SPOT-6 / 7
- Pleiades 1

Users can also define the sensor of their choice, either by providing central wavelength and full width at half maximum (fwhm) corresponding to each band and assuming gaussian response for each band, or by providing a file describing the exact SRF.

```
# get the spectral response function (SRF) for Sentinel-2A
srf_s2 <- get_srf_sensor(sensor_name = 'Sentinel_2A')

# get the SRF corresponding to a sensor defined by user.
# Hyperspectral sensor: 10 nm spectral sampling & 10 nm fwhm for all bands
wl <- seq(400, 2500, by = 10)
fwhm <- rep(x = 10, length = length(wl))
spectral_props = data.frame(wl = wl, fwhm = fwhm)
sensor_name <- 'Hyperspectral_Sensor'
srf_hsi <- get_srf_sensor(sensor_name = sensor_name,
                                        spectral_properties = spectral_props)
```

PROSAIL inversion through iterative optimization

Iterative optimization with no prior information

PROSAIL is a relatively simple and computationally efficient model. As with PROSPECT, inversion based on iterative optimization can be considered. However, unlike PROSPECT inversion which is well-posed in most situations, PROSAIL inversion is inherently ill-posed. Therefore, standard iterative optimization algorithms consisting in minimizing a cost function will converge towards local minimum, unless global optimization algorithms are applied. This means that initial conditions of the iterative optimization influence the output of a local optimization.

Multiple inversion strategies based on iterative optimization are implemented in the package `prospect`. The main iterative optimization algorithm is based on the minimization of a multivariable function with nonlinear constraints. This procedure is based on the function `fmincon` included in the package `pracma`. The default criterion for the cost function corresponds to the root mean square error (RMSE) between a reference surface reflectance and surface reflectance simulated with PROSAIL. The code below provides an example of iterative optimization applied to simulated surface reflectance at native spectral sampling (1 nm), and to Sentinel-2 surface reflectance.

```
# define initial value, lower and upper bounds for inversion
init <- data.frame('chl' = 40, 'car' = 10, 'ewt' = 0.01, 'lma' = 0.01,
                   'lai' = 3, 'lidf_a' = 50, 'n_struct' = 1.5)
lb <- data.frame('chl' = 5, 'car' = 1, 'ewt' = 0.002, 'lma' = 0,
                 'lai' = 0.5, 'lidf_a' = 30, 'n_struct' = 1)
ub <- data.frame('chl' = 80, 'car' = 20, 'ewt' = 0.03, 'lma' = 0.03,
                 'lai' = 6, 'lidf_a' = 80, 'n_struct' = 3)

# define parameters set for inversion
parm_set <- data.frame('tts' = 40, 'tto' = 0, 'psi' = 60, 'soil_brightness' = 1,
                       'ant' = 0, 'brown' = 0, 'hotspot' = 0.1)

# define soil reflectance (set as prior knowledge) and spec_soil variable
rsoil <- parm_set$soil_brightness*spec_soil_ossl$soil_01
spec_soil <- data.frame('lambda' = spec_soil$lambda,
                        'refl' = rsoil)

# simulate canopy surface reflectance with 1 nm sampling
truth <- data.frame('chl' = 60, 'car' = 8, 'ewt' = 0.015, 'lma' = 0.005,
                    'lai' = 3,  'lidf_a' = 60, 'n_struct' = 1.8)
```

```
refl_1nm <- prosail(n_struct = truth$n_struct, chl = truth$chl, car = truth$car,
                    ant = parm_set$ant, brown = parm_set$brown,
                    ewt = truth$ewt, lma = truth$lma, type_lidf = 2,
                    lai = truth$lai, hotspot = parm_set$hotspot,
                    lidf_a = init$lidf_a, rsoil = rsoil, tts = parm_set$tts,
                    tto = parm_set$tto, psi = parm_set$psi)
surf_refl_1nm <- get_surf_refl(rdot = refl_1nm$rdot, rsot = refl_1nm$rsot,
                               tts = parm_set$tts,
                               spec_atm_sensor = spec_atm)

# invert PROSAIL on surface reflectance with 1 nm spectral sampling
est_1nm <- invert_prosail(refl_mes = surf_refl_1nm$surf_refl,
                          initialization = init,
                          lower_bound = lb, upper_bound = ub, type_lidf = 2,
                          spec_prospect_sensor = spec_prospect_full_range,
                          spec_atm_sensor = spec_atm,
                          spec_soil_sensor = spec_soil,
                          parm_set = parm_set)

# convert spectral input based on Sentinel-2 SRF
srf_s2 <- get_srf_sensor(sensor_name = 'Sentinel_2A')
spec_prospect <- spec_prospect_full_range
lambda <- spec_prospect_full_range$lambda
surf_refl_S2 <- apply_sensor_characteristics(wvl = lambda,
                                             srf = srf_s2,
                                             refl = surf_refl_1nm$surf_refl)
spec_prospect_s2 <- apply_sensor_characteristics(wvl = lambda, srf = srf_s2,
                                                 refl = spec_prospect)
spec_soil_s2 <- apply_sensor_characteristics(wvl = lambda, srf = srf_s2,
                                             refl = spec_soil)
spec_atm_s2 <- apply_sensor_characteristics(wvl = lambda, srf = srf_s2,
                                            refl = spec_atm)

# invert PROSAIL on Sentinel-2 surface reflectance
est_s2 <- invert_prosail(refl_mes = surf_refl_S2, initialization = init,
                         lower_bound = lb, upper_bound = ub,
                         spec_prospect_sensor = spec_prospect_s2,
                         spec_atm_sensor = spec_atm_s2,
                         spec_soil_sensor = spec_soil_s2,
                         type_lidf = 2, parm_set = parm_set)
```


Iterative optimization with prior information

Prior information can be added to the cost function in order to regularize the ill-posed inverse problem (Combal et al., 2003). Multiple possibilities exist to account for prior information:

- the definition of `initialization` variable according to initial assumptions
- the addition of a term corresponding to the probability density function of part or all of the parameters to be estimated in the cost function.

The code blow illustrates the introduction of prior information on LIDF, with the definition of a probability density function corresponding to the average leaf angle `lidf_a`. A weighting factor associated to the relative importance of this prior information with respect to the main criterion to minimize (RMSE between reference surface reflectance and simulated surface reflectance) can also be adjusted.

```r
# define prior information on average leaf angle lidf_a
prior_info <- list('mean' = data.frame('lidf_a' = 60),
                   'sd' = data.frame('lidf_a' = 10),
                   'weight_prior' = 0.01)

# invert PROSAIL on surface reflectance with 1 nm spectral sampling and
# probability density function for lidf_a
est_1nm_p <- invert_prosail(refl_mes = surf_refl_1nm$surf_refl,
                            initialization = init,
                            lower_bound = lb, upper_bound = ub,
                            spec_prospect_sensor = spec_prospect_full_range,
                            spec_atm_sensor = spec_atm,
                            spec_soil_sensor = spec_soil,
                            type_lidf = 2, parm_set = parm_set,
                            prior_info =  prior_info)

# invert PROSAIL on Sentinel-2 surface reflectance and probability density
# function for lidf_a
est_s2_p <- invert_prosail(refl_mes = surf_refl_S2, initialization = init,
                           lower_bound = lb, upper_bound = ub,
                           spec_prospect_sensor = spec_prospect_s2,
                           spec_atm_sensor = spec_atm_s2,
                           spec_soil_sensor = spec_soil_s2,
                           type_lidf = 2, parm_set = parm_set,
                           prior_info =  prior_info)
```

PROSAIL hybrid inversion

Hybrid model inversion is defined here as a two-step inversion procedure:

- run RTM to produce a simulated LUT including sensor surface reflectance and corresponding PROSAIL input parameters, including vegetation properties.
- train a machine learning regression algorithm to estimate a vegetation property or a set of vegetation properties of interest from sensor surface reflectance.

The next subsections describe the strategy followed by the hybrid inversion implemented in `prosail`.

Simulating surface reflectance to prepare for regression model training

Several factors need to be accounted for when preparing for the training stage of the regression models :

- the sampling strategy (range, distribution, co-distributions for input parameters)
- the addition of additive and multiplicative noise to the simulated surface reflectance
- the definition of spectral bands to be used for the training of the regression model

The default parameterization of `prosail` hybrid inversion relies on the production of a surface reflectance LUT compliant with the specifications defined in the ATBD for the Biophysical Processor of the Sentinel toolbox (Weiss et al., 2020). However, each step of the simulation of a training LUT can be defined by user.

Training a machine learning regression algorithm

The machine learning strategy implemented in `prosail` consists in a parsimonious ensemble method based on a bootstrap aggregating (bagging) prediction of biophysical properties from

multiple support vector regression (SVR) models. Each SVR model is trained with a limited number of samples, ensuring fast training stage. The predicted value then corresponds to the mean prediction from a set of SVR models. The standard deviation can also be derived from this ensemble of predictors. However, standard deviation may not provide accurate model uncertainty quantification (Palmer et al., 2022).

The default parameterization of this inversion strategy uses an ensemble of 10 SVR models, each of them trained with 200 samples. Therefore, only 2000 samples are required for the training stage, which is relatively low compared to the 41472 cases used for the training of the artificial neural networks used in the SNAP toolbox. This parsimonious ensemble method ensures very fast training stage and similar retrieval performances as those obtained with the SNAP toolbox.

Performing full hybrid inversion

Figure 1 summarizes the workflow applied to perform hybrid inversion. The function `train_prosail_inversion` combines all aforementioned steps to produce surface reflectance simulations and train SVR models. It provides regression models as outputs, which can then be directly used to estimate vegetation biophysical properties from data tables or raster data.

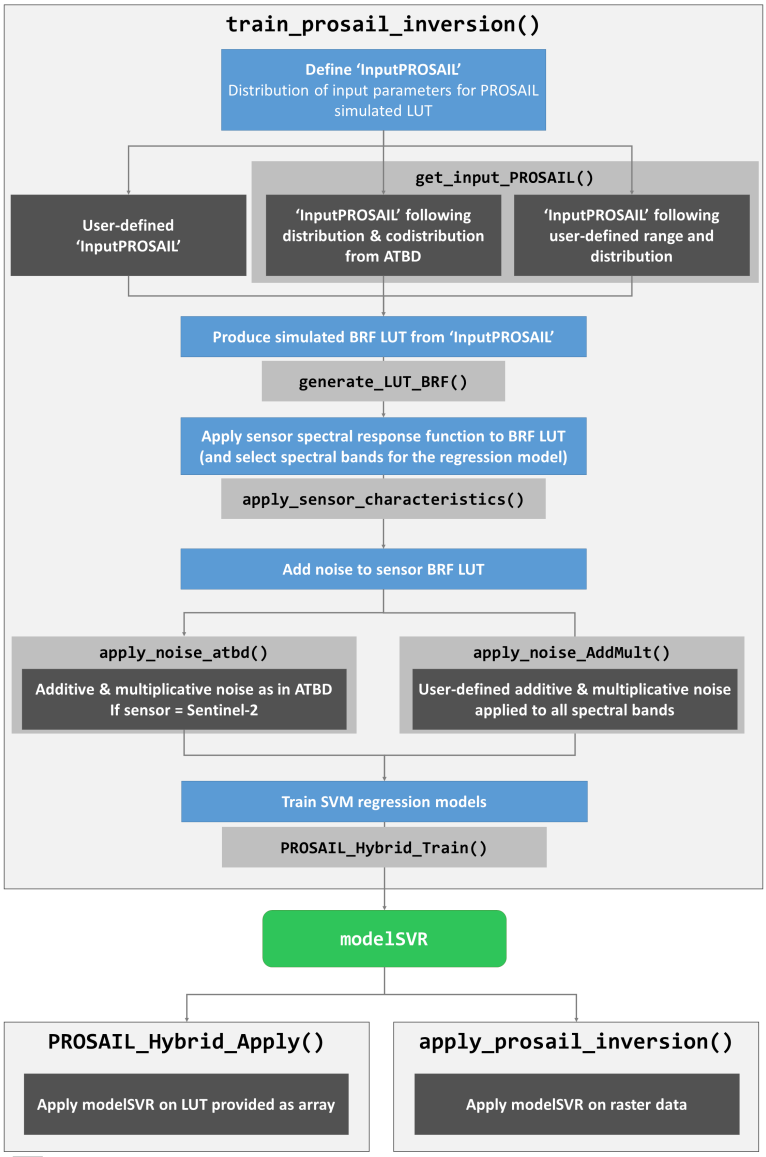

Figure 1: Workflow of PROSAIL hybrid inversion implemented in the package prosail.

The example code below illustrates the training stage with the function `train_prosail_inversion`, to estimate LAI, fCover and fAPAR from Sentinel-2, using the three spectral bands corresponding to green channel (B3), red channel (B4) and near infrared channel (B8) as specified in the ATBD document.

```
# define sensor to simulate with PROSAIL
srf <- get_srf_sensor(sensor_name = 'Sentinel_2')

# define parameters to estimate
parms_to_estimate <- c('lai', 'fcover', 'fapar')

# define spectral bands required to train SVR model for each variable
selected_bands <- list()
for (parm in parms_to_estimate)
  selected_bands[[parm]] <- c('B3','B4','B8')
```

```r
# define output directory where LUTs will be saved
prosail_dir <- './hybrid_inversion'

# define ranges for geometry of acquisition
geom_acq <- list('min' = data.frame('tto' = 0, 'tts' = 20, 'psi' = 0),
                 'max' = data.frame('tto' = 10, 'tts' = 30, 'psi' = 360))

# train model
modelSVR <- train_prosail_inversion(parms_to_estimate = parms_to_estimate,
                                    atbd = TRUE, geom_acq = geom_acq,
                                    srf = srf, selected_bands = selected_bands,
                                    output_dir = prosail_dir)
```

The processing steps called in `train_prosail_inversion` can be broken down as follows:

```r
# produce input variables following ATBD specifications and geometry of acq.
input_prosail <- get_input_prosail(atbd = TRUE, geom_acq = geom_acq)

# generate a LUT with 1 nm spectral sampling
res <- generate_lut_prosail(SAILversion = '4SAIL',
                            input_prosail = input_prosail,
                            spec_prospect = spec_prospect_full_range,
                            spec_soil = spec_soil_atbd_v2, spec_atm = spec_atm)

# include fcover and fAPAR (derived from SAIL reflectance / absortion factors)
# in the pool of variables to explain
input_prosail$fcover <- res$fcover
input_prosail$fapar <- res$fapar

# apply Sentinel-2 SRF to get sensor surface reflectance
surf_refl_LUT <- apply_sensor_characteristics(wvl = lambda, srf = srf,
                                              refl = res$surf_refl)
# identify spectral bands in LUT
rownames(surf_refl_LUT) <- srf$spectral_bands

# apply noise and produce a LUT for each parameter
surf_refl_noise <- list()
for (parm in parms_to_estimate)
  surf_refl_noise[[parm]] <- apply_noise_atbd(refl_lut = surf_refl_LUT)

# train a set of SVR models for each parameter
modelSVR <- list()
for (parm in parms_to_estimate){
  input_variables <- input_prosail[[parm]]
  modelSVR[[parm]] <- prosail_hybrid_train(refl_lut = surf_refl_noise[[parm]],
                                           input_variables = input_variables)
}
```

The resulting models can then be applied on data tables or raster data. The following example perform predictions based on models trained previously. The prediction returns mean value obtained form the ensemble of regression models for each sample, as well as corresponding standard deviation.

```r
mean_estimate <- sd_estimate <- list()
for (parm in parms_to_estimate){
  HybridRes <- prosail_hybrid_apply(regression_models = modelSVR[[parm]],
```

```
                                        refl = surf_refl_noise[[parm]])
  mean_estimate[[parm]] <- HybridRes$mean_estimate
  sd_estimate[[parm]] <- HybridRes$sd_estimate
}
```

Application to imagery data

Here, we illustrate how to run `prosail` hybrid inversion on Sentinel-2 imagery. Sentinel-2 imagery handling requires the installation of `preprocs2`, a package dedicated to downloading and preprocessing of Sentinel-2 data from different providers. It also requires the package `sf` to handle vector data.

Here, for the sake of comparison between biophysical variables produced from SNAP and those produced with the `prosail` hybrid inversion, the Sentinel-2 Level-2A product corresponding to tile *30SWJ* acquired on *May 13th, 2021* was downloaded from the Copernicus Data Space Ecosystem (CDSE). This is important for the consistency of the processing baseline of Sentinel-2 acquisitions, as the processing baseline used by alternative Sentinel-2 data providers such as Microsoft Planetary Computers and Google Cloud may not be consistent with the latest version provided by CDSE.

Users are invited to go through the documentation of the package `prosail`, which provides examples on how to download Sentinel-2 imagery data from specific regions of interest instead of downloading a full SAFE tile.

Once the SAFE product downloaded `preprocs2` is used to crop and save reflectance data corresponding to an area of interest defined within the Sentinel-2 tile footprint.

```
# load preprocS2 and sf
library(preprocS2)
library(sf)
# define path for original image subset and prosail outputs
main_dir <- './barrax'                              # main directory
safe_dir <- file.path(main_dir, 's2_safe')          # SAFE directory
out_s2 <- file.path(main_dir, 'S2_subset')          # S2 subset directory
out_BP <- file.path(main_dir, 'prosail_inversion')  # prosail products directory

# search for L2A scene of interest with sen2proc
stac_items <- s2_search(start = "2021-05-13", level = "L2A", tile = "30SWJ")

# download files with sen2proc
dir.create(path = safe_dir, showWarnings = FALSE, recursive = TRUE)
safe_path <- cdse_download(x = stac_items, outdir = safe_dir)

# define area of interest included in S2 tile
aoi_bbox <- st_bbox(obj = c('xmin' = 571626, 'ymin' = 4324941,
                            'xmax' = 582995, 'ymax' = 4331253))
aoi <- bbox_to_poly(aoi_bbox, crs = 32630)
path_aoi <- file.path(main_dir, 'barrax.gpkg')
st_write(obj = aoi, dsn = path_aoi, driver = 'GPKG', delete_dsn = T)

# extract area of interest from SAFE with preprocS2
s2_products <- extract_from_safe(safe_path = safe_path,
                                 path_aoi = path_aoi,
                                 output_dir = out_s2)
raster_path <- s2_products$Refl_path
mask_path <- s2_products$cloudmasks$BinaryMask
```

The resulting raster data can then be directly processed with `prosail` using the code provided hereafter.

```
# get geometry of acquisition with preprocS2
s2_geom <- get_s2_geometry(MTD_TL_xml = s2_products$MTD_TL)
geom_acq <- list('min' = data.frame('tto' = min(s2_geom$VZA, na.rm = T),
                                    'tts' = min(s2_geom$SZA, na.rm = T),
                                    'psi' = min(abs(s2_geom$SAA-s2_geom$VAA),
                                                na.rm = T)),
                 'max' = data.frame('tto' = max(s2_geom$VZA, na.rm = T),
                                    'tts' = max(s2_geom$SZA, na.rm = T),
                                    'psi' = max(abs(s2_geom$SAA-s2_geom$VAA),
                                                na.rm = T)))
# get sensor response for Sentinel-2
srf <- get_srf_sensor(sensor_name = 'Sentinel_2B')
band_names <- srf$spectral_bands

# define parameters to estimate
parms_to_estimate <- c('lai', 'fcover', 'fapar', 'chl')

# define spectral bands required to train SVR model for each variable
selected_bands <- list('lai' = c('B3','B4','B8'),
                       'fcover' = c('B3','B4','B8'),
                       'fapar' = c('B3','B4','B8'),
                       'chl' = c('B3','B4','B5','B6','B7','B8'))

# train SVR model for each variable of interest
modelSVR <- train_prosail_inversion(parms_to_estimate = parms_to_estimate,
                                    atbd = TRUE, geom_acq = geom_acq,
                                    srf = srf,
                                    selected_bands = selected_bands,
                                    output_dir = out_BP)

# apply SVR on S2 L2A reflectance
options <- set_options_prosail(fun = 'apply_prosail_inversion')
options$multiplying_factor <- 10000
options$tiling <- TRUE
BPvars <- apply_prosail_inversion(raster_path = raster_path,
                                  mask_path = mask_path,
                                  hybrid_model = modelSVR,
                                  output_dir = out_BP,
                                  band_names = band_names,
                                  selected_bands = selected_bands,
                                  options = options)
```

Comparison with SNAP toolbox

LAI, fcover and fAPAR computed from Sentinel-2 images with the hybrid inversion implemented in `prosail` are compared to those produced with SNAP in Figure 2.

Biophysical properties estimated with `prosail` vs Biophysical properties estimated with SNAP.

Figure 2: Biophysical properties estimated with `prosail` vs Biophysical properties estimated with SNAP.

The two methods show relatively good consistency with Pearson correlation coefficient ranging

between 0.99 and 1.00 for LAI, and superior to 0.97 for fCover and fAPAR. Algorithmic differences remain between the results obtained from the biophysical processor of the SNAP toolbox and the inversion obtained with `prosail`, including: - the version of the PROSPECT model: SNAP uses a version anterior to PROSPECT-D, - the machine learning algorithm.

The differences in the retrieval of the vegetation biophysical properties suggest differences in the soil properties accounted for in simulations with low LAI, despite our efforts to reproduce the workflow described in the ATBD (Weiss et al., 2020). Additional tests performed over other sites (on both croplands and forests) showed similar performances.

The availability of open source and fully parameterizable inversion procedures should contribute to improve reproducibility of currently available softwares.

Conclusion

We introduce `prosail`, an R package dedicated to the canopy reflectance model PROSAIL, coupling PROSPECT and SAIL. `prosail` is coupled with the R package `prospect` in order to allow seamless integration of future versions of the leaf model. It allows simulation of bi-directional reflectance factor and surface reflectance for any type of optical sensor based on their spectral response, including multispectral sensors and hyperspectral sensors. The package also includes a collection of inversion procedures, including iterative optimization with and without prior information, and hybrid inversion.

The estimation of vegetation biophysical properties with `prosail` hybrid inversion is consistent with estimations produced with the method implemented in the Sentinel toolbox, which is currently the reference method.

`prosail` hybrid inversion does not intend to be computationally as efficient as SNAP and may not meet requirements for large scale vegetation monitoring applications. `prosail` hybrid inversion offers a fully adjustable hybrid inversion framework, including an original parsimonious machine learning regression method, allowing fast and efficient training for any type of sensor. It is a valuable tool for experimenting on the potential and limitation of physically-based retrieval of vegetation traits from various types of optical sensors, including airborne imaging spectroscopy, as well satellite missions already operational and in preparation.

Availability

`prosail` is an open-source software package made available under the MIT license. Tutorials are available at https://jbferet.gitlab.io/prosail/.

Acknowledgements

The authors acknowledge financial support from Agence Nationale de la Recherche (BioCop project — ANR-17-CE32-0001). We are grateful to Wout Verhoef for the development of the initial version of the 4SAIL and 4SAIL2 models. We are grateful to Stéphane Jacquemoud and Frédéric Baret for the development of the initial version of the prospect model.

AI Usage Disclosure

No generative AI tools were used in the development of this software, the writing of this manuscript, or the preparation of supporting materials.

References

Berger, K., Atzberger, C., Danner, M., D'Urso, G., Mauser, W., Vuolo, F., & Hank, T. (2018). Evaluation of the PROSAIL model capabilities for future hyperspectral model environments: A review study. *Remote Sensing*, *10*(1). https://doi.org/10.3390/rs10010085

Bréon, F.-M., Maignan, F., Leroy, M., & Grant, I. (2002). Analysis of hot spot directional signatures measured from space. *Journal of Geophysical Research: Atmospheres*, *107*(D16), AAC 1-1-AAC 1-15. https://doi.org/10.1029/2001JD001094

C. François, C. Ottlé, A. Olioso, L. Prévot, N. Bruguier, & Y. Ducros. (2002). Conversion of 400-1100 nm vegetation albedo measurements into total shortwave broadband albedo using a canopy radiative transfer model. *Agronomie*, *22*(6), 611–618. https://doi.org/10.1051/agro:2002033

Caicedo, J. P. R., Verrelst, J., Muñoz-Marí, J., Moreno, J., & Camps-Valls, G. (2014). Toward a semiautomatic machine learning retrieval of biophysical parameters. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *7*(4), 1249–1259. https://doi.org/10.1109/JSTARS.2014.2298752

Campbell, G. S. (1990). Derivation of an angle density function for canopies with ellipsoidal leaf angle distributions. *Agricultural and Forest Meteorology*, *49*(3), 173–176. https://doi.org/10.1016/0168-1923(90)90030-A

Combal, B., Baret, F., Weiss, M., Trubuil, A., Macé, D., Pragnère, A., Myneni, R., Knyazikhin, Y., & Wang, L. (2003). Retrieval of canopy biophysical variables from bidirectional reflectance: Using prior information to solve the ill-posed inverse problem. *Remote Sensing of Environment*, *84*(1), 1–15. https://doi.org/10.1016/S0034-4257(02)00035-4

Féret, J.-B., Berger, K., Boissieu, F. de, & Malenovský, Z. (2021). PROSPECT-PRO for estimating content of nitrogen-containing leaf proteins and other carbon-based constituents. *Remote Sensing of Environment*, *252*, 112173. https://doi.org/10.1016/j.rse.2020.112173

Féret, J.-B., & Boissieu, F. de. (2024). Prospect: An r package to link leaf optical properties with their chemical and structural properties with the leaf model PROSPECT. *Journal of Open Source Software*, *9*(94), 6027. https://doi.org/10.21105/joss.06027

Féret, J.-B., Boissieu, F. de, & Lang, M. (2026). A plant trait-based approach to map forest biodiversity using imaging spectroscopy and physical modeling. In D. Rocchini (Ed.), *R coding for ecology* (pp. 161–182). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-99665-8_7

Féret, J.-B., Gitelson, A. A., Noble, S. D., & Jacquemoud, S. (2017). PROSPECT-D: Towards modeling leaf optical properties through a complete lifecycle. *Remote Sensing of Environment*, *193*, 204–215. https://doi.org/10.1016/j.rse.2017.03.004

Féret, J.-B., le Maire, G., Jay, S., Berveiller, D., Bendoula, R., Hmimina, G., Cheraiet, A., Oliveira, J. C., Ponzoni, F. J., Solanki, T., de Boissieu, F., Chave, J., Nouvellon, Y., Porcar-Castell, A., Proisy, C., Soudani, K., Gastellu-Etchegorry, J.-P., & Lefèvre-Fonollosa, M.-J. (2019). Estimating leaf mass per area and equivalent water thickness based on leaf optical properties: Potential and limitations of physical modeling and machine learning. *Remote Sensing of Environment*, *231*, 110959. https://doi.org/10.1016/j.rse.2018.11.002

Ferreira, V. B., Féret, J.-B., Atzberger, C., Guillemot, J., Campoe, O., & Maire, G. le. (2026). Enhancing PROSAIL inversion: Key considerations and practical insights. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *19*, 1378–1390. https://doi.org/10.1109/JSTARS.2025.3635001

Gastellu-Etchegorry, J.-P., Yin, T., Lauret, N., Cajgfinger, T., Gregoire, T., Grau, E., Féret, J.-B., Lopes, M., Guilleux, J., Dedieu, G., Malenovský, Z., Cook, B. D., Morton, D., Rubio,

J., Durrieu, S., Cazanave, G., Martin, E., & Ristorcelli, T. (2015). Discrete anisotropic radiative transfer (DART 5) for modeling airborne and satellite spectroradiometer and LIDAR acquisitions of natural and urban landscapes. *Remote Sensing*, *7*(2), 1667–1701. https://doi.org/10.3390/rs70201667

Hauser, L. T., Féret, J.-B., An Binh, N., van der Windt, N., Sil, Â. F., Timmermans, J., Soudzilovskaia, N. A., & van Bodegom, P. M. (2021). Towards scalable estimation of plant functional diversity from sentinel-2: In-situ validation in a heterogeneous (semi-)natural landscape. *Remote Sensing of Environment*, *262*, 112505. https://doi.org/10.1016/j.rse.2021.112505

Hijmans, R. J. (2025). *Terra: Spatial data analysis*. https://CRAN.R-project.org/package=terra

Jacquemoud, S., & Baret, F. (1990). PROSPECT: A model of leaf optical properties spectra. *Remote Sensing of Environment*, *34*(2), 75–91. https://doi.org/10.1016/0034-4257(90)90100-Z

Jacquemoud, S., Verhoef, W., Baret, F., Bacour, C., Zarco-Tejada, P. J., Asner, G. P., François, C., & Ustin, S. L. (2009). PROSPECT+SAIL models: A review of use for vegetation characterization. *Remote Sensing of Environment*, *113*, S56–S66. https://doi.org/10.1016/j.rse.2008.01.026

Kuusk, A. (1991). The hot spot effect in plant canopy reflectance. In R. B. Myneni & J. Ross (Eds.), *Photon-vegetation interactions: Applications in optical remote sensing and plant ecology* (pp. 139–159). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-75389-3_5

Palmer, G., Du, S., Politowicz, A., Emory, J. P., Yang, X., Gautam, A., Gupta, G., Li, Z., Jacobs, R., & Morgan, D. (2022). Calibration after bootstrap for accurate uncertainty quantification in regression models. *Npj Computational Materials*, *8*(1), 115. https://doi.org/10.1038/s41524-022-00794-8

Poulter, B., Currey, B., Calle, L., Shiklomanov, A. N., Amaral, C. H., Brookshire, E. N. J., Campbell, P., Chlus, A., Cawse-Nicholson, K., Huemmrich, F., Miller, C. E., Miner, K., Pierrat, Z., Raiho, A. M., Schimel, D., Serbin, S., Smith, W. K., Stavros, N., Stutz, J., … Zhang, Z. (2023). Simulating global dynamic surface reflectances for imaging spectroscopy spaceborne missions: LPJ-PROSAIL. *Journal of Geophysical Research: Biogeosciences*, *128*(1), e2022JG006935. https://doi.org/10.1029/2022JG006935

Spafford, L., le Maire, G., MacDougall, A., de Boissieu, F., & Féret, J.-B. (2021). Spectral subdomains and prior estimation of leaf structure improves PROSPECT inversion on reflectance or transmittance alone. *Remote Sensing of Environment*, *252*, 112176. https://doi.org/10.1016/j.rse.2020.112176

Spitters, C. J. T., Toussaint, H. A. J. M., & Goudriaan, J. (1986). Separating the diffuse and direct component of global radiation and its implications for modeling canopy photosynthesis part i. Components of incoming radiation. *Agricultural and Forest Meteorology*, *38*(1), 217–229. https://doi.org/10.1016/0168-1923(86)90060-2

Verhoef, W. (1984). Light scattering by leaf layers with application to canopy reflectance modeling: The SAIL model. *Remote Sensing of Environment*, *16*(2), 125–141. https://doi.org/10.1016/0034-4257(84)90057-9

Verhoef, W. (1998). *Theory of radiative transfer models applied in optical remote sensing of vegetation canopies : E-book* [PhD Thesis - Research external, graduation external]. Landbouwuniversiteit Wageningen (LUW). ISBN: 90-5485-804-4

Verhoef, Wout, & Bach, H. (2007). Coupled soil–leaf-canopy and atmosphere radiative transfer modeling to simulate hyperspectral multi-angular surface reflectance and TOA radiance data. *Remote Sensing of Environment*, *109*(2), 166–182. https://doi.org/10.1016/j.rse.

2006.12.013

Verhoef, W., Jia, L., Xiao, Q., & Su, Z. (2007). Unified optical-thermal four-stream radiative transfer theory for homogeneous vegetation canopies. *IEEE Transactions on Geoscience and Remote Sensing*, *45*(6), 1808–1822. https://doi.org/10.1109/TGRS.2007.895844

Verrelst, J., Camps-Valls, G., Muñoz-Marí, J., Rivera, J. P., Veroustraete, F., Clevers, J. G. P. W., & Moreno, J. (2015). Optical remote sensing and the retrieval of terrestrial vegetation bio-geophysical properties – a review. *ISPRS Journal of Photogrammetry and Remote Sensing*, *108*, 273–290. https://doi.org/10.1016/j.isprsjprs.2015.05.005

Weiss, M., Baret, F., & Jay, S. (2020). *S2ToolBox Level 2 products LAI, FAPAR, FCOVER* [Research Report]. EMMAH-CAPTE, INRAe Avignon. https://hal.inrae.fr/hal-03584016

Yang, P., Prikaziuk, E., Verhoef, W., & Tol, C. van der. (2021). SCOPE 2.0: A model to simulate vegetated land surface fluxes and satellite signals. *Geoscientific Model Development*, *14*(7), 4697–4712. https://doi.org/10.5194/gmd-14-4697-2021