PDinsight user manual

Eva Deinum

July 30, 2024

1 General information

PDinsight is a python program for predicting tissue (cell-cell) level effective permeabilities for symplasmic transport from ultrastructural (EM) data. A typical use would be to compare these values to tissue level assessments of effective permeabilities based on photobleaching / photoactivation of carboxyfluorecein (CF) (Rutschow et al., 2011) or suitable GFP derivates (e.g., DRONPA-s (Gerlitz et al., 2018)). The first version of PDinsight was released along with "From plasmodesma geometry to effective symplasmic permeability through biophysical modelling", Deinum et al. eLife 2019. Mathematical notation in this manual follows (Deinum et al., 2019). If using PDinsight in a publication, please cite this manuscript. If using the `singleInterface` and/or `bootstrapInterface` modes for using raw data, please also cite (Deinum, 2022)

PDinsight is written in python 3. If available, it uses the numpy module, but does not strictly depend upon that. The program has different modes for computing the parameter requirements for a given effective symplasmic wall permeability $P(\alpha)$ for particles of radius α and related quantities. The different modes and the relevant parameters are controlled from a parameter file (default: parameters.txt). A graphical user interface (GUI) is provided to help the user create parameter files and run PDinsight. The GUI is written using TKinter, which is included in standard installation of python.

For electron microscopists, who typically have access to many ultrastructural parameters, but often do not know $P(\alpha)$, the mode `computeVals` will be useful. This computes the expected $P(\alpha)$ values when taking all parameters at face value. Comparison with the sub-nano channel model is possible. In principle, all model parameters must be defined, but missing parameters may be explored using lists of possible values or left at a default (e.g., cell length L and a triangular distribution of PDs), as these have little influence on $P(\alpha)$. If estimates of PD density and distribution are missing, the mode `computeUnitVals`, which computes $\Pi(\alpha)$ (a "unit permeability", i.e., assuming a density of 1 PD/μm^2 and $f_{ih} = 1$), could be useful. In this mode, comparison with the sub-nano channel is impossible.

In tissue level experiments, $P(\alpha)$ is typically measured, but not all ultrastructural parameters will be known. It is likely that PD density ρ and radius (R_n, assuming straight channels) are poorly known. In this case, mode `computeRnDensityGraph` will be useful, or a combination of modes `computeDens` and `computeAperture` and a number of guesses for $R_n/\bar{\alpha}$ (maximum α that would fit through the channel) or ρ, respectively. Additional poorly known parameters can be explored as suggested above. If uncertain, it is strongly advised to explore PD length l ($\approx$cell wall thickness). For thick cell walls ($l \geq 200$ nm), it may be worth exploring the effects of increased central radius ($R_c > R_n$). This is currently only possible in modes `computeVals` and `computeUnitVals`.

PDinsight is published under the GNU general public licence v3.0.

2 Modes

Major modes The core of PDinsight is computing effective permeabilities ($P(\alpha)$) for symplasmic transport based on all model parameters mentioned in the manuscript. This is in mode `computeVals`. The same computations are used in other modes, which compute the requirements for obtaining a given target value (or set of values) of $P(\alpha)$. In mode `computeDens`, required densities (ρ) are computed for given values of maximum particle size $\bar{\alpha}$ (and other parameters). In mode `computeAperture`, required apertures, given as $\bar{\alpha}$ as well as neck radius R_n, are computed for given values of ρ (and other parameters). In mode `computeRnDensityGraph`, R_n, ρ curves are computed that together yield a target $P(\alpha)$. The corresponding values of $\bar{\alpha}$ are also reported. These curves can be visualized using any plotting program. Mode `sensitivityAnalysis` computes so-called elasticities (normalized partial derivatives) around a given set (or sets) of parameters. These elasticities tell how sensitive calculated values of $P(\alpha)$, $\Pi(\alpha)$ and constituents like f_{ih} (a correction factor for the fact that the cell wall is only symplasmically permeable where the PDs are) are on the parameters involved.

(New in version 2): In mode `singleInterface`, $P(\alpha)$ is computed based on raw data, using each entry (set of data for one PD) exactly once. In mode `bootstrapInterface`, $P(\alpha)$ is computed based on raw data, resampling `bootstrapSamples` times (with replacement) to a sample size of `bootstrapSingleSampleSize` if specified, or the number of PDs in the original data set.

Auxillary modes In computing $P(\alpha)$, correction factor f_{ih} is automatically included. For specific cases such as modelling studies, however, it may be useful to calculate f_{ih} separately. For this purpose, several modes exist for exploring inhomogeneity factor f_{ih}: `computeFih_subNano` (function of $\bar{\alpha}$; also output values for sub-nano model), `computeFih_pitField_dens` (function of ρ), `computeFih_pitField_xMax` (function of $\bar{\alpha}$) and `computeTwinning` (function of ρ_{pits}, cluster density).

By default, computations are performed for the unobstructed sleeve model (Deinum et al., 2019). Most computations can also be performed for the sub-nano channel model Liesche and Schulz (2013); Comtet et al. (2017). Using switch `compSubNano`, values for the sub-nano model are also computed.

3 Graphical user interface

The GUI to PDinsight is written to facilitate the creation of parameter files and also has a button to run PDinsight directly based on the parameters displayed. In contrast to the generic parameter file, the GUI only shows fields for parameters that are actually used in a specific run mode. The first step of using the gui is to select the run mode (choose from: `computeVals`, `computeUnitVals`, `computeDens`, `computeAperture`, `computeRnDensityGraph` and `sensitivityAnalysis`), followed by "load default parameters". This overwrites any user input from previous modes. All required parameters are shown as text entry fields, with radio buttons for the relevant options. Basic validation occurs on the fly (valid input type, etc). When done, click "Run PDinsight" to write a parameter file and run the program or "Export parameter file" to write a parameter file only. Note that switching modes requires clicking "load default parameters", i.e., a fresh start.

Additional information can be obtained by clicking the "Info" button and, for certain parameters, clicking on the parameter name. If additional information is available, the mouse cursor changes into a question mark.

4 Command line usage

Command line usage (linux): python PDinsight.py PARAMETERFILE. (windows): first open a command prompt window (e.g., by searching for "cmd" in the search bar) and go to the directory where PDinsight is located. Then type: py PDinsight.py PARAMETERFILE

Using the GUI from the command line (linux): python PDinsightGUI.py or (windows): py PDinsightGUI.py The correct version of PDinsight.py should also be available in the current directory.

Note that it is possible to run multiple modes in a single run by setting the respective values behind the options to 1, provided their parameters are fully compatible. For example (based on version 2+):

```
singleInterface 1
bootstrapInterface 1
```

5 Output files

By default, all outputs are tab separated files (.tsv). This can be switched to csv using the option “outputType” (options “tsv” and “csv”).

6 Program maintenance

The latest version of PDinsight + documentation can be downloaded from Github: `https://github.com/eedeinum/PDinsight` It is recommended to update both the core (PDinsight.py) and the GUI (PDinsightGUI.py) simultaneously. If, for whatever reason you would like to update only the core, please note that you have to adapt the following line in the GUI by hand:

```
import PDinsight as pd
```

7 Model parameters

See Figure 1 for an overview of single channel PD parameters and PD geometry. See Table 1 for a list of model parameters (PDinsight names) and the corresponding mathematical symbols used in Deinum et al. (2019). See Table 3 for available modes and Table 4 for available options. See Table 2 for the parameters specific to `singleInterface` and `bootstrapInterface` (version 2+).

References

Comtet J, Turgeon R, Stroock A. Phloem loading through plasmodesmata: a biophysical analysis. Plant physiology. 2017; p. pp–01041.

Deinum EE. More Insights from Ultrastructural and Functional Plasmodesmata Data Using PDinsight. In: *Plasmodesmata* Springer; 2022.p. 443–456. doi: 10.1007/978-1-0716-2132-5˙30.

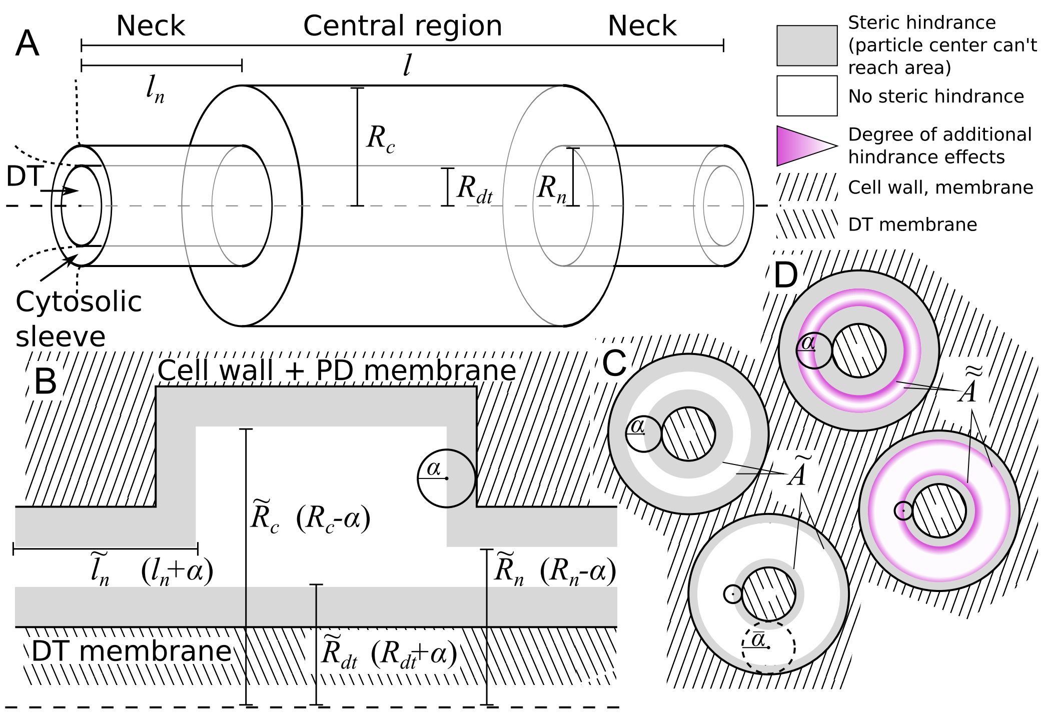
A
Neck
Central region
Neck
l_n
l
R_c
R_{dt}
R_n
DT
Cytosolic sleeve
Steric hindrance (particle center can't reach area)
No steric hindrance
Degree of additional hindrance effects
Cell wall, membrane
DT membrane
B
Cell wall + PD membrane
α
$\widetilde{R}_c$ $(R_c$-$\alpha)$
$\widetilde{l}_n$ $(l_n+\alpha)$
$\widetilde{R}_n$ $(R_n$-$\alpha)$
DT membrane
$\widetilde{R}_{dt}$ $(R_{dt}+\alpha)$
C
D
$\widetilde{A}$
$\widetilde{\widetilde{A}}$
$\bar{\alpha}$

Figure 1: Overview of single channel PD parameters. Figure from Deinum et al. (2019). A: structural parameters as used for input (PD dimensions). B-D: Cross sections showing how internally, available volumes are adjusted for steric hindrance (B,C) and additional (hydrodynamic) hindrance effects (D) depending on particle size α. A particle with radius $\bar{\alpha}$ would touch both channel walls in the narrowest (neck) region.

Table 1: List of parameters and mathematical symbols (Deinum et al., 2019). [List] indicates that (depending on mode), a list of values may be entered.

PDinsight		User unit	Description
x	α	nm	Particle radius
xMax	$\bar{\alpha}$	nm	Maximum particle radius that fits through a (model) PD
diff	D	μm^2/s	*Particle size dependent* diffusion constant
Fih	f_{ih}	-	Correction factor for inhomogeneous wall permeability ($0 \leq f_{ih} \leq 1$)
Lneck[List], Lneck2[List]	l_n	nm	Neck length
Lpd	l	nm	Total PD length
Lcell	L	μm	Cell length
Rc[List]	R_c	nm	Central region radius
Rdt[List]	R_{dt}	nm	DT radius
Rn[List], Rn2[List]	R_n	nm	Neck radius
dens[List]	ρ	PDs /μm^2	PD density
pit[List]	p	-	number of PDs per cluster ("pit field"). Options: 1, 2, 3, 4, 5, 6, 7, 12, 19.
dPit[List]	p	nm	nearest neighbour distance between PDs inside pit field.
Peff	$P(\alpha)$	μm/s	symplasmic permeability (for particles of size α) of the entire cell wall
Punit	$\Pi(\alpha)$	μm/s	symplasmic permeability (for particles of size α) of a single PD per unit of cell wall surface, without correction factor f_{ih} ($\Pi(\alpha) = \frac{P(\alpha)}{f_{ih}\rho}$).
grid[List]			PDs (or pit fields) are assumed to spaced on a grid. Options: "triangular", "square", "hexagonal" or "hex" and "random".
fileTag			Prefix for all output files.

Table 2: List of parameters specific for `singleInterface` and `bootstrapInterface`.

PDinsight	Default	Description
noData	-1	token for missing data.
randomSeed	-1	Random seed (if positive). With negative numbers, no random seed is specified.
bootstrapSamples	10000	Number of resampling iterations when bootstrapping.
bootstrapSingleSampleSize	0	Number of PDs per resampled sample. When 0, PD number is the same as the original data.
DTintercept	Rdt	for specifying linear DT model: Rdt = DTintercept + varA*coefA [+ varB*coefB + ...]
DTvarList		for specifying linear DT model (see above). Must be same length as DT-coefList.
DTvarCoef		for specifying linear DT model (see above). Must be same length as DT-varList.

Table 3: List of modes. Modes indicated with * are only available through the command line interface.

Name	Description
computeRnDensityGraph	Compute a graph of R_n, ρ pairs that matches a given $P(\alpha)$ given structural parameters etc.
computeDens	Compute required density ρ given a (list of) $\bar{\alpha}$ values.
computeAperture	Compute required PD aperture (and related $\bar{\alpha}$) given a (list of) ρ values.
computeVals	Compute $P(\alpha)$ given structural parameters etc.
computeUnitVals	Compute $\Pi(\alpha)$ given single PD structural parameters only.
sensitivityAnalysis	Compute elasticities (normalized partial derivatives) around a given set (or sets) of parameters.
singleInterface	Compute $P(\alpha)$ given raw data with measurements per PD.
bootstrapInterface	Compute $P(\alpha)$ confidence intervals using a bootstrapping approach. Requires raw data with measurements per PD.
$*$ computeFih_subNano	Compute f_{ih}, including comparison with sub-nano channel model, for $\bar{\alpha} \in [2\alpha, 50]$ nm.
$*$ computeFih_pitField_xMax	Compute f_{ih}, possibly for multiple values of p, for $\bar{\alpha} \in [2\alpha, 50]$ nm.
$*$ computeFih_pitField_dens	Compute f_{ih}, possibly for multiple values of p, for $\rho \in [0.1, 30]$ PDs/μm^2.
$*$ computeTwinning	Compute f_{ih}, possibly for multiple values of p, where the total density is ρp and $\rho \in [0.1, 30]$ PDs/μm^2.

Table 4: List of options

Name	Description
compSubNano	Compare with sub-nano channel model.
computeClusterIncrease	If true, dens indicates cluster density rather than total density. If false, dens indicates total density.
printRn	(Additionally) include R_n in output. For some modes, R_n is always printed.
doNotCombine	If true, "list" parameters must contain either 1 or n entries, where n is the same for all. PDinsight uses the i^{th} entry (or single value) of each list as a set. Useful for evaluating raw data. If false, PDinsight evaluates all possible combinations of entries in all lists.
asymmetricPDs	Allow for different neck dimensions on either side of the PD. If true, Rn2[List] and Lneck2[list] are required.
outputType	tsv or csv
PDmodel	1cyl (straight), 2cyl (different Rn and Rc), 3cyl (different Rn, Rn2 and Rc). (version 2+)
DTformula	Allows for a linear model for DT radius rather than a fixed value, e.g., based on statistical model (version 2+)

Deinum EE, Mulder BM, Benitez-Alfonso Y. From plasmodesma geometry to effective symplasmic permeability through biophysical modelling. Elife. 2019; 8:e49000. doi: 10.7554/eLife.49000.

Gerlitz N, Gerum R, Sauer N, Stadler R. Photoinducible DRONPA-s: a new tool for investigating cell–cell connectivity. The Plant Journal. 2018; 94(5):751–766.

Liesche J, Schulz A. Modeling the parameters for plasmodesmal sugar filtering in active symplasmic phloem loaders. Front Plant Sci. 2013; 4:207. `http://dx.doi.org/10.3389/fpls.2013.00207`, doi: 10.3389/fpls.2013.00207.

Rutschow HL, Baskin TI, Kramer EM. Regulation of solute flux through plasmodesmata in the root meristem. Plant Physiol. 2011 Apr; 155(4):1817–1826. `http://dx.doi.org/10.1104/pp.110.168187`, doi: 10.1104/pp.110.168187.