Chapter 29

Computational Approaches for Peroxisomal Protein Localization

Marco Anteghini and Vitor A. P. Martins dos Santos

Abstract

Computational approaches are practical when investigating putative peroxisomal proteins and for sub-peroxisomal protein localization in unknown protein sequences. Nowadays, advancements in computational methods and Machine Learning (ML) can be used to hasten the discovery of novel peroxisomal proteins and can be combined with more established computational methodologies. Here, we explain and list some of the most used tools and methodologies for novel peroxisomal protein detection and localization.

Key words Subcellular localization, Sub-organelle localization, Peroxisome targeting signal, Cellular compartments, Machine learning

1 Introduction

Advancements in organelle-specific research are possible also thanks to the use of case-specific tools such as for sub-peroxisomal and sub-mitochondrial protein localization [1–5]. These tools, nowadays easily accessible and user-friendly, allow researchers to perform fast and accurate screening while looking for new peroxisomal and mitochondrial proteins [1–5]. Alternatively, general methods for protein sequence localization can be handy if re-adapted for specific use cases [6–11].

For example, a general peroxisomal protein search from a given set of FASTA sequences can start by detecting the predicted subcellular localization using DeepLoc-2.0 [6]. After filtering for predicted peroxisomal proteins, a researcher can either look for known Peroxisomal Targeting Signals (PTSs) to further filter the dataset [3] or retrieve a list of candidates for future analysis or experimental validations [12].

PTSs are consensus motifs found in many peroxisomal proteins. Specific receptors recognize a PTS and bind to a region of

Michael Schrader (ed.), *Peroxisomes: Methods and Protocols*, Methods in Molecular Biology, vol. 2643,
https://doi.org/10.1007/978-1-0716-3048-8_29,

the peroxisomal protein [13]. The known PTSs are: (1) PTS1. The PTS1 receptor is encoded by the PEX5 gene [14] and the PTS1 is defined as the final dodecamer with a focus on the terminal tripeptide [15]; (2) PTS2. It is an N-terminal targeting signal and its receptor is encoded by the PEX7 gene (a co-receptor is also involved in the peroxisomal import) [16]; (3) mPTS. It is a cis-acting targeting signal specific for peroxisomal membrane proteins. Its mechanism is still poorly understood [17]. The algorithms defined in Schlüter et al. [3] can detect these different PTSs, and the PTS1 can now be easily and accurately detected also on the OrganelX web server (https://organelx.hpc.rug.nl/fasta/compute_in_pts), as described in recent works [1, 5].

In this chapter, we list a number of practical tools accompanied by specific use cases and a workflow on how to perform a complete peroxisomal protein localization search. A service bundle and a practical study example [18] support the workflow presented here.

2 Materials

2.1 Use Case-Specific Tools (See Note 1)

1. **PeroxisomeDB**. The PEROXISOME DATABASE (PeroxisomeDB) organizes and integrates curated information about peroxisomes. That includes genes, proteins, molecular functions, metabolic pathways, and their related disorders [3]. Related prediction tools are also available at http://www.peroxisomedb.org/. In the scope of this chapter, we report three main tools for different PTSs detection: (1) PTS1 binding sites; (2) PTS2 binding sites; (3) Pex19BS binding sites. All the three programs rely on multiple sequence alignments where the input sequence or the input BLOCK is aligned with a predefined BLOCK that contains a specific category of proteins (e.g. proteins containing PTS1).
2. **In-Pero**. A computational pipeline that discriminates between matrix and membrane proteins [1]. In-Pero relies on a Support Vector Machine classifier trained on the statistical representation of protein sequences obtained by combining two deep-learning embeddings (UniRep + SecVec) [19, 20]. In-Pero can be executed locally following the instruction available at https://github.com/MarcoAnteghini/In-Pero or on the dedicated web server available at https://organelx.hpc.rug.nl/fasta/compute_in_pero.
3. **In-Mito**. A computational pipeline that allows classifying the four sub-mitochondrial compartments: matrix, internal-membrane, inter-membrane space, and external membrane [1] (*see* **Note 2**) In-Mito relies on a Support Vector Machine classifier trained on the statistical representation of protein

sequences obtained by combining two deep-learning embeddings (UniRep + SecVec) [19, 20]. In-Mito can run locally following the instruction available at https://github.com/MarcoAnteghini/In-Mito or on the dedicated web server available at https://organelx.hpc.rug.nl/fasta/compute_in_mito.

4. **DeepMito**. A computational method for predicting sub-mitochondrial localization based on a convolutional neural network architecture [2] (*see* **Note 2**). Based on a given input protein, DeepMito can discriminate the four sub-mitochondrial compartments: matrix, internal-membrane, inter-membrane space, and external membrane. DeepMito is available at http://busca.biocomp.unibo.it/deepmito/.

2.2 General Tools for Subcellular Localization and Transmembrane Detection (See Note 1)

1. **TMHMM2.0** and **DeepTMHMM**. TMHMM2.0 is a membrane protein topology prediction method based on a hidden Markov model (HMM) [8]. It predicts transmembrane helices and discriminates between soluble and membrane proteins. The tool is available at https://services.healthtech.dtu.dk/service.php?TMHMM-2.0.

   DeepTMHMM is a novel version of the TMHMM predictor. It is the most complete and currently the best-performing method for the membrane protein topology prediction [21]. The model encodes the primary amino acid sequence by a pre-trained language model and decodes the topology by a state-space model to produce topology and type predictions. DeepTMHMM is available at https://dtu.biolib.com/DeepTMHMM.

2. **Phobius**. Combined transmembrane topology and signal peptide predictor [11]. The predictor relies on a HMM that models the different sequence regions of a signal peptide and the different regions of a transmembrane protein in a series of interconnected states. Phobius is available at https://phobius.sbc.su.se/

3. **DeepLoc-2.0** [6]. Multi-localization prediction tool based on a pretrained protein language model that uses a three-stage deep learning approach for sequence classification. (1) The feature representation for each amino acid in the sequence is generated. (2) Attention-based pooling stage produces a single representation for the whole sequence. (3) The prediction stage uses a classifier to output the subcellular labels. DeepLoc-2.0 is available at https://services.healthtech.dtu.dk/service.php?DeepLoc-2.0

4. **PSORT**. A computer program that predicts protein localization sites in cells and its last version is WoLF PSORT [7]. WoLF PSORT converts protein amino acid sequences into numerical localization features; based on sorting signals, amino acid

composition and functional motifs. A k-nearest neighbor classifier is used for the final prediction. The webserver is available at https://psort.hgc.jp/

5. **TargetP-2.0**. Deep Learning method to identify N-terminal sorting signals, which direct proteins to the secretory pathway, mitochondria, and chloroplasts or other plastids [10]. The method relies on Bi-directional Recurrent Neural Networks (BiRNN) with Long short-term memory (LSTM) cells and a multi-attention mechanism [22]. TargetP-2.0 is available at https://services.healthtech.dtu.dk/service.php?TargetP-2.0.

3 Methods

3.1 Detection of Peroxisomal Protein Candidates

The workflow for a typical analysis represented as a service bundle is shown in Fig. 1 and also available (with functional links for each tool) at https://tess.elixir-europe.org/workflows/peroxisomal-candidates-detection. For an accurate analysis, it is recommended to first look for known PTSs when available and then proceed with further filtering steps. After the PTS detection, we can investigate the presence of transmembrane regions in the amino acid sequence and filter the results according to the detected PTS. In particular, we can exclude membrane proteins while checking for PTS1 or PTS2. Afterwards, if stringent filtering is required, it is recommended to analyze the candidates with other subcellular localization tools (see Subheading 2.2) and remove the proteins with unexpected predicted localization.

Alternatively, we can start our analysis directly from the subcellular localization prediction and then run the predicted peroxisomal proteins with a sub-peroxisomal classification tool that does not consider PTS motifs [1, 5]. As shown in Fig. 1, after the subcellular localization prediction, if we obtain mitochondrial proteins, it is possible to either run DeepMito or In-Mito, while we can execute In-Pero for the peroxisomal proteins [1, 2]. These tools discriminate the sub-organelle compartments, which are four in the case of mitochondria (matrix, internal-membrane, inter-membrane space, external membrane) and two in the case of peroxisomes (matrix, membrane) [1, 2].

As a final step for further validation, selected sequences can be screened for conservation of the potential PTS1/PTS2 using BLAST (the last version while writing this chapter is BLAST+ 2.13.0) [23, 24].

An example of a complete pipeline can be found in the recent work of Kamoshita et al. [18] analyzing the peroxisomal protein inventory of zebrafish. For simplicity, we report here a summarized computational pipeline: (1) The *Danio rerio* proteome was downloaded from UniProt (https://www.uniprot.org/) [25] and

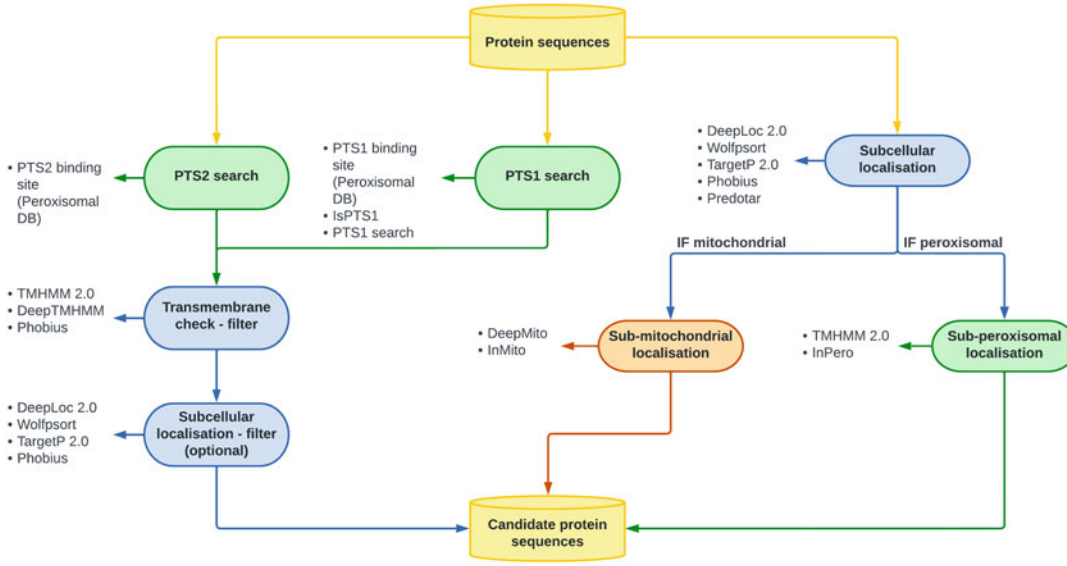

Fig. 1 Workflow and Service Bundle of a standard peroxisomal and mitochondrial protein localization analysis. The workflow starts with the initial dataset containing protein sequences in FASTA format. The starting point is the root of the graph "Protein sequences". Each node represents a step of the analysis. Its associated tools are visible on the left of the node. The workflow converges in the final node "Candidate protein sequences," where candidate protein sequences are selected for future analyses or experimental validation

screened for proteins carrying a PTS1 at the very C-terminus matching the consensus motif [ASCNPHTG]-[RKHQNSL]-[LMIVF]; (2) Among 46,848 proteins, 2638 proteins matching the pattern were identified and filtered for non-membrane proteins with TMHMM Server v. 2.02 [8] (1966 protein left); (3) The 1966 protein sequences were further analyzed with WoLF PSORT (Package Command Line Version 0.2) [7] and entries with Endoplasmic Reticulum as possible subcellular localization were removed (1171 sequences left); (4) The identified proteins were further analyzed by PTS1 predictor algorithms [3] and sequences which produced no hit with the "metazoa" or "general" modus of the software were removed (371 proteins left); (5) Finally, the obtained entries were manually curated, integrating information from Zebrafish-specific datasets and considered for experimental validation.

4 Notes

1. Most of the tools presented in this chapter are designed for eukaryotes. Some of them can be used for prokaryotic organisms as well (e.g. DeepTMHMM [21]). Note that peroxisomes are only present in eukaryotes. We advise the user to check the specifications of each tool in the original web server or publication.

2. In this chapter, we list some of the available tools for mitochondrial protein detection. Important components of the organelle division machinery present a dual localization (peroxisomal and mitochondrial). Moreover, both organelles have proven to be in continuous interplay [26]. For an accurate peroxisomal protein localization search, it is advised to look into mitochondrial localization too.

References

1. Anteghini M, Martins dos Santos VAP, Saccenti E (2021) In-Pero: exploiting deep learning Embeddings of protein sequences to predict the localisation of Peroxisomal proteins. Int J Mol Sci 22(12):6409
2. Savojardo C, Bruciaferri N, Tartari G et al (2019) DeepMito: accurate prediction of protein sub-mitochondrial localization using convolutional neural networks. Bioinformatics 36(1):56–64
3. Schlüter A, Real-Chicharro A, Gabaldón T et al (2009) PeroxisomeDB 2.0: an integrative view of the global peroxisomal metabolome. Nuc Acid Res 38:D800–D805
4. Claros MG, Vincens P (1996) Computational method to predict Mitochondrially imported proteins and their targeting sequences. Eur J Biochem 241(3):779–786
5. Anteghini M, Haja A, Martins dos Santos VAP et al (2022) OrganelX web server for sub-peroxisomal and sub-mitochondrial protein localisation. bioRxiv. https://doi.org/10.1101/2022.06.21.497045
6. Thumuluri V, Almagro Armenteros JJ, Rosenberg Johansen A et al (2022) DeepLoc 2.0: multi-label subcellular localization prediction using protein language models. Nuc Acid Res Apr 30, gkac278. https://doi.org/10.1093/nar/gkac278
7. Horton P, Park K-J, Obayashi T et al (2007) WoLF PSORT: protein localization predictor. Nuc Acid Res 35:W585–W587
8. Krogh A, Larsson B, von Heijne G, Sonnhammer ELL (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. J Mol Biol 305(3):567–580
9. Small I, Peeters N, Legeai F, Lurin C (2004) Predotar: a tool for rapidly screening proteomes for N-terminal targeting sequences. Proteomics 4(6):1581–1590
10. Almagro Armenteros JJ, Salvatore M, Emanuelsson O et al (2019) Detecting sequence signals in targeting peptides using deep learning. Life Sci Alliance 2(5):e201900429
11. Käll L, Krogh A, Sonnhammer ELL (2004) A combined transmembrane topology and signal peptide prediction method. J Mol Biol 338(5):1027–1036
12. Schrader TA, Islinger M, Schrader M (2017) Detection and Immunolabeling of Peroxisomal proteins. Methods Mol Biol 1595:113–130
13. Gould SG, Keller GA, Subramani S (1987) Identification of a peroxisomal targeting signal at the carboxy terminus of firefly luciferase. J Cell Biol 105(6):2923–2931
14. Kiel JAKW, Emmrich K, Meyer HE, Kunau WH (2005) Ubiquitination of the peroxisomal targeting signal type 1 receptor, Pex5p, suggests the presence of a quality control mechanism during peroxisomal matrix protein import. J Biol Chem 280(3):1921–1930
15. Brocard C, Hartig A (2006) Peroxisome targeting signal 1: is it really a simple tripeptide? Biochim Biophys Acta - Mol Cell Res 1763(12):1565–1573
16. Kunze M (2020) The Type-2 peroxisomal targeting signal. Biochim Biophys Acta, Mol Cell Res 1867(2):118609
17. Van Ael E, Fransen M (2006) Targeting signals in Peroxisomal membrane proteins. Biochim Biophys Acta, Mol Cell Res 1763(12):1629–1638
18. Kamoshita M, Kumar R, Anteghini M et al (2022) Insights into the Peroxisomal protein inventory of zebrafish. Front Phys 13:822509
19. Alley E, Khimulya G, Biswas S et al (2019) Unified rational protein engineering with sequence-based deep representation learning. Nat Methods 16(12):1315–1322
20. Heinzinger M, Elnaggar A, Wang Y et al (2019) Modeling aspects of the language of life through transfer-learning protein sequences. BMC Bioinform 20(1):723
21. Hallgren J, Tsirigos KD, Pedersen MD et al (2022) DeepTMHMM predicts alpha and Beta transmembrane proteins using deep neural networks. bioRxiv. https://doi.org/10.1101/2022.04.08.487609

22. Lin Z, Feng M, Nogueira dos Santos C et al (2017) A Structured self-attentive sentence embedding. arXiv Preprint arXiv:1703.03130
23. Altschul SF, Gish W, Miller W et al (1990) Basic local alignment search tool. J Mol Biol 215(3):403–410
24. Camacho C, Coulouris G, Avagyan V et al (2009) BLAST+: architecture and applications. BMC Bioinform 10(1):421. https://doi.org/10.1186/1471-2105-10-42125
25. Consortium, UniProt T (2020) UniProt: the universal protein knowledgebase in 2021. Nuc Acid Res 49(D1):D480–D489
26. Schrader M, Costello JL, Godinho LF, Islinger M (2015) Peroxisome-mitochondria interplay and disease. J Inherit Metab Dis 38(4): 681–702