SuperHirn User Manual

Author:	Lukas Mueller
Address:	Institute for Molecular Systems Biology ETH Hönggerberg, HPT C 75 Wolfgang Pauli-Str. 16 CH-8093 Zürich, Switzerland
Email:	Lukas.Mueller@imsb.biol.ethz.ch

Content:

1. Introduction
2. Installation
3. Overview of program workflow
4. Program running parameters
5. Making your data accessible to *SuperHirn*
6. *SuperHirn*'s modular functionalities
7. Data storage and data output
8. Example data set
9. Further reading, references and links

1. Introduction

SuperHirn is an open source program written in C++ for the quantitative analysis of liquid chromatography mass spectrometry (LC-MS) data. More detailed algorithmic description can be found in Mueller et al (Mueller, 2006) or in the software applications (Bodenmiller, et al., 2006; Rinner and Mueller, et al., 2006). The program is executed from command line and program parameters are read from parameter text files. *SuperHirn* is built in a modular manner reflecting the different LC-MS processing functionalities. In particular, the workflow of the program comprises following modules:

- Feature extraction of LC-MS runs and integration of MS2 peptide identifications
- LC-MS similarity assessment
- Multiple LC-MS alignment to generate a *MasterMap*
- Intensity normalization of MS1 features in the *MasterMap*
- MS1 feature annotation of *MasterMap*
- Unsupervised feature profiling by *Kmeans* clustering
- Targeted peptide/protein profiling

More details about these modules and their functionality are provided in section 6.

2. Installation

SuperHirn can be downloaded free of charge from:

http://tools.proteomecenter.org/SuperHirn.php

The program can be compiled on Linux and OSX platform, however only slight changes in the header files of the source code should render SuperHirn compatible to Windows platforms. *SuperHirn* uses the free available tinyxml *C++* library (http://sourceforge.net/projects/tinyxml/), which is incorporated into the source distribution of *SuperHirn* so that no external libraries are required.
The source code is organized into two main folders:

CPP_LIBRARY: a *C++* class library offering different proteomics functionality. This library is independent of *SuperHirn* and we encourage developers to use this code for their own applications. We will provide a detailed description of the different classes in the near future.

SuperHirn: this folder contains the source code, which links the functionalities of the *CPP_LIBRARY* classes. In addition, the folder contains the main root parameter file ROOT_PARAM.def that stores a stable set of program parameters (for details on the program parameters see section 4). The make file to compile *SuperHirn* is also found in the subfolder "make".

Compilation of *SuperHirn*:
SuperHirn is compiled from command line. The executable will be saved into the folder "make". If you would like to change this location, edit the make variable "EXE_NAME" in the make file. To compile proceed as following:

1. move to the directory "SuperHirn/make/"
2. type "make all"

(to clean up everything: "make clean")

SuperHirn can now be executed. To see a list of program running options (or see the detailed description in section 5 and 6) type:

./SuperHirn –h

3. Overview of the program workflow

The main principle of *SuperHirn* is to preprocess raw profile LC-MS runs by a MS1 feature extraction routine and to annotate theses features with available MS2 peptide identifications. Section 5 provides details how to make the raw data (profile raw LC/MS runs and MS2 peptide interpretations) available to the program and what formats are supported. The preprocessed and MS2 containing LC-MS runs are then combined into a *MasterMap* by the multiple LC/MS alignment. The *MasterMap* contains all required information for the post processing steps. The computational steps of *SuperHirn* are classified into two main data processing groups:

PreProcessing: a.) MS1 feature extraction and integration of MS2 data followed by LC-MS similarity analysis
b.) Multiple LC-MS alignment to a *MasterMap*

Post Processing: c.) Intensity normalization of the *MasterMap*
d.) Feature profiling by Kmeans clustering
e.) Targeted peptide/protein profiling analysis
f.) MS1 feature annotation of the *MasterMap*

While steps *a* and *b* are required to construct the *MasterMap*, the other steps *c-f* are optional. These steps are fully performed on the constructed *MasterMap* and their usage depends on the analysis focus of the users.

4. Parameter file for *SuperHirn*

Root parameter set:
SuperHirn extracts the parameters for the different processing routines from a text based parameter file. There are many parameters, which render *SuperHirn* flexible to work with different LC-MS data types or enable to change the program's capability to different applications. *SuperHirn* contains a stable set of parameters in the root parameter file "SuperHirn/make/ROOT_PARAM.def". Therefore the user only has to care about parameters that she/he is really interested in and all other parameters will be automatically read from the ROOT_PARAM.def. The parameters of this file are currently optimized for high mass precision MS data and should not be modified unless date from low mass accuracy instruments are processed (>10ppm).

Personal parameter set:
There is a basic set of parameters that have to be defined by the user. These user-defined parameters are stored in a file called param.def, which has to be located in the folder where *SuperHirn* is executed. It is recommended to use the example param.def file in "SuperHirn/example/param.def" as a starting point to create a personal parameter file. If you would like or need to modify other program parameter, please copy these parameters from the ROOT_PARAM.def into your personal parameter file. All parameters in your personal param.def file will overwrite the root parameters. More information about some basic parameters in the param.def will follow in section 5.

5. *Program data accessibility*

Please use the example parameter file (SuperHirn/example/param.def) as a starting point to create your personal parameter file. This file contains the basic parameters to run *SuperHirn*, which you have to adopt.

MY PROJECT NAME: name of your project. SuperHirn will create for a folder ANALYSIS_ *MY PROJECT NAME* and store all the data in this location.

MZXML DIRECTORY: location of the folder containing the raw profile LC-MS data. *SuperHirn* is compatible to mzXML formatted LC-MS runs (Pedrioli, et al., 2004). To generate mzXML files from native MS acquisition files, try on of the converters from the sashimi website (http://sashimi.sourceforge.net). **PLEASE NOTE that the mzXML data have to be acquired in PROFILE mode!!!!**

PEPXML DIRECTORY: location of the folder containing your MS2 peptide assignments. Currently, *SuperHirn* offers support for pepXML formatted MS2 peptide assignments (Keller, et al., 2005).

Retention time tolerance: assumed measurement error of your LC system (min.)

MS1 m/z tolerance: mass to charge precision of your MS instrument on MS1 level

MS2 m/z tolerance: mass to charge precision of your MS instrument on MS2 level

MS2 matching modus: you can use either the measured precursor mass or the theoretical mass of a peptide assignment to find its corresponding MS1 feature

MS2 scan tolerance: scan tolerance to assign a MS2 peptide identification to the elution peak of a MS1 features.

6. SuperHirn program modules

SuperHirn contains the following 6 modules for the analysis of LC-MS data. The modules are grouped into 2 categories:

i.) these modules have to be executed in a distinct order to create a *MasterMap*
ii.) optional independent *MasterMap* analysis modules

The following text will explain how to use these modules and in what order they should be performed.

i. Modules to construct the *MasterMap*

These modules should be executed in their listed order to create a *MasterMap*:

a. Build the alignment tree: this module contains different functionalities. Initially, it performs a MS1 feature extraction routine on all the raw profile input LC-MS (**formatted in mzXML and acquired in profile mode**) and then associates available MS2 information to the extracted MS1 features. In the next step, pairwise LC-MS similarity analysis is performed to construct a similarity tree of the LC-MS input data, which is used in the next module *b* for the multiple LC-MS alignment process.

Command: *SuperHirn -BT*

b. Multiple LC-MS alignment: based on step *a*, the input LC-MS runs are combined into a *MasterMap* by a multi dimensional LC-MS alignment process. The *MasterMap* is then stored in APML format (Refr) and also in a tab delimited text format (see section 7).

Command: *SuperHirn –CM*

ii. Modules for the *MasterMap* analysis

These modules are executed optionally after the construction of the *MasterMap* and depend on the users data analysis interests:

c. MasterMap intensity normalization: normalizes MS1 feature intensities across all LC-MS runs stored in the *MasterMap*. The normalized *MasterMap* is then stored in a text file called "normalized_MasterMap.txt". Please note that intensity values in the APML formatted *MasterMap* are not modified.

Command: *SuperHirn -IN*

d. Unsupervised MS1 feature profiling by Kmeans: performs an unsupervised *Kmeans* clustering analysis of MS1 feature profiles in the *MasterMap*.

Command: *SuperHirn -DP*

e. Targeted peptide/protein profiling: this module is based on the previous *Kmeans* clustering analysis of the *MasterMap* (step *d*). From the constructed *Kmeans* clusters, the feature members of the closest cluster to a user defined input target profile are selected. These features are assembled into peptides and proteins and their consensus profile correlation to the target profile is evaluated.

Command: *SuperHirn -EME*

f. MS1 feature annotation of the MasterMap: the MasterMap can be further updated by MS2 information in order to assign peptide identifications to MS1 features which have not been annotated in the current LC-MS experiment. A folder needs to be defined in the param.def file (*INCLUSION LIST DIRECTORY*), where the additional MS2 peptide identification data is located (for example from other MS instruments or from inclusion list runs). This data is then integrated into the *MasterMap* by searching for every new MS2 peptide identification its corresponding MS1 feature in the *MasterMap*.

Command: *SuperHirn -ILA*

7. Data storage

All data is stored into a folder named according your project name defined in the param.def file. Please find a detailed description of the different files that SuperHirn creates in the section 8 where the example data set is described.

8. Example data set

To offer a starting point for the user, we provide an example data set, which can be download from the program website (*http://tools.proteomecenter.org/SuperHirn.php*). The data set consists of 3 preprocessed LC/MS runs, each one is a sample from a drosophila total cell lysates. The runs were searched against the drosophila fly base (flybase-r4.3) by SORCERER-SEQUEST (TM) v3.0.3 search algorithm run on the SageN Sorcerer machine. This data set contains also an example param.def file, which can then be adapted to other data sets. The only thing you will need to change in this files is the path of ROOT_PARAM.def file. Please adjust this parameter to the correct path (its in the SuperHirn/make folder).

Data Folder structure and the data files:

- Analysis_SuperHirn_Example:
    - LC_MS_RUN: preprocessed LC/MS run in APML
    - LC_MS_XML_similiarity_matrix.xml: XML format of the LC-MS similarity scores
    - PROCESSED_MASTER.xml: *MasterMap* in APML format
    - MASTER_RUN.txt: *MasterMap* in text format
    - my_guide_tree.tre: *alignment topology* in tre-format

-SEARCHES: contains pepXML formatted MS2 peptide identifications

9. Web links and References

Web:

Superhirn download page:	*http://tools.proteomecenter.org/SuperHirn.php*
TinyXML:	*http://sourceforge.net/projects/tinyxml/*
Sashimi (mzXML converters):	*http://tools.proteomecenter.org/*

References:

Bodenmiller, B., Mueller, L.N., Müller, M. and Aebersold, R. (2006) Isolation of Distinct, Overlapping Segments of the Phospho-Proteome, *Nature Methods*, **accepted**.

Keller, A., Eng, J., Zhang, N., Li, X.J. and Aebersold, R. (2005) A uniform proteomics MS/MS analysis platform utilizing open XML file formats, *Mol Syst Biol*, **1**, 2005 0017.

Mueller, L.N. et al. (2006) SuperHirn - a novel tool for high resolution LC-MS based pep-tide/protein profiling, *in preparation*.

Pedrioli, P.G., Eng, J.K., Hubley, R., Vogelzang, M., Deutsch, E.W., Raught, B., Pratt, B., Nilsson, E., Angeletti, R.H., Apweiler, R., Cheung, K., Costello, C.E., Hermjakob, H., Huang, S., Julian, R.K., Kapp, E., McComb, M.E., Oliver, S.G., Omenn, G., Paton, N.W., Simpson, R., Smith, R., Taylor, C.F., Zhu, W. and Aebersold, R. (2004) A common open representation of mass spectrometry data and its application to proteomics research, *Nat Biotechnol*, **22**, 1459-1466.

Rinner, O., Mueller, L.N., Hubalek, M., Müller, M., Gstaiger, M. and Aebersold, R. (2006) MasterMap: an integrated mass spectrometric and computational framework for the comprehensive analysis of protein interaction networks, *submitted*.