

( BiNoM )<sup>n</sup>



# **BiNoM: A Biological Network Manager**

## **Manual**

Eric Bonnet, Laurence Calzone, Daniel Rovera, Gautier Stoll, Emmanuel Barillot, Andrei Zinovyev

Team Computational Systems Biology of Cancer  
Institut Curie  
26, rue d’Ulm  
75005 Paris

## **Contents**

# 1 Introduction

BiNoM (BIological NetWOrk Manager) is a Cytoscape plugin, developed to facilitate the manipulation of biological networks represented in standard systems biology formats and to carry out studies on the network structure. BiNoM provides the user with a complete interface for the analysis of biological networks in Cytoscape environment.

In an effort to exchange and curate pathway database knowledge, several standard formats have been developed (SBML, BioPAX [?] and others). Many softwares, which are centered on the description and representation of biological pathways, adopted these standards. CellDesigner[?] and Cytoscape[?], for instance, allow the visualization and manipulation of networks but meet some limitations. BiNoM was designed to facilitate the use of systems biology standards, the extraction and organization of information from pathway databases through BioPAX interface.

BiNoM concentrates on the following aspects: the import and export of BioPAX and (CellDesigner) SBML files and the conversion between them; the structural analysis of biological networks including decomposition of networks into modules, path analysis, etc.; the BioPAX query engine which provides the extraction of information from huge BioPAX files such as whole pathway databases; and various operations on graphs not offered by Cytoscape such as clipboard operations and comparison of networks.

BiNoM plugin with documentation, API and source code is available for download (<http://binom.curie.fr>).

## 2 BiNoM I/O

### 2.1 Import BioPAX 3 document

BioPAX level 3 information is fully supported (reaction network, interaction network, pathway structure, annotations).

#### Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Import BioPAX 3 Document from file

The model M-Phase-L3.owl [?] is uploaded. A dialog window proposes to create three different interfaces from the BioPAX file: reaction network (RN), pathway structure (PS) and interaction map (IM).

- Reaction network: M-Phase-L3 RN is a representation of the reaction network (figure ??).
- Pathway structure: M-Phase-L3 PS represents the pathway hierarchical structure. For this example, we choose to show a more detailed and complete pathway, the apoptosis sub-network extracted from Reactome database (figure ??).

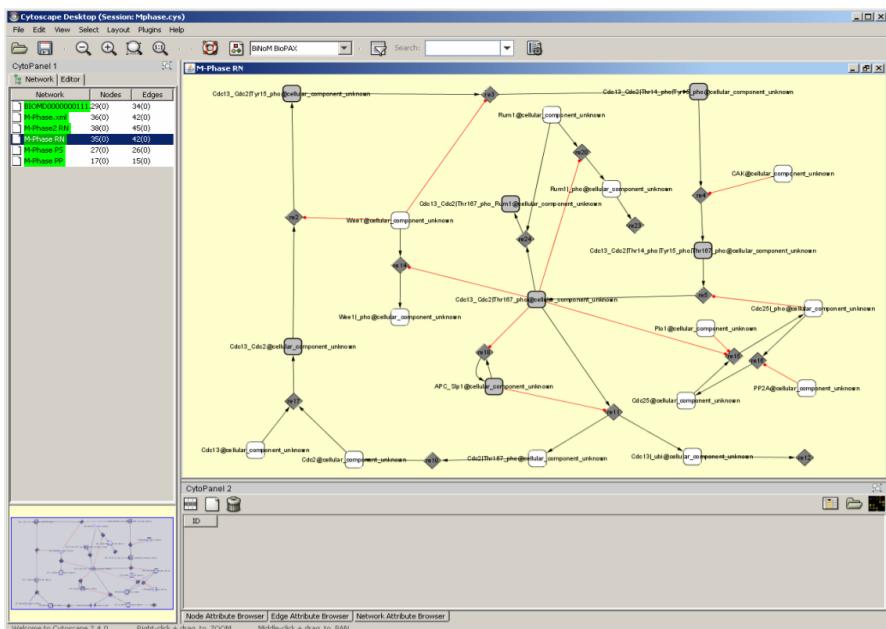


Figure 1: BioPAX view of Novak et al. model.

- Protein interaction: M-Phase-L3 IM shows which proteins interact with each other.

For more details on BioPAX, its interfaces, etc, go to section ??.

In the case of creating the pathway structure interface, several choices are offered:

- Make Root Pathway Node: adds an extra node to which all pathways are connected. This feature can be useful for organizing the graph and joining separate and disjoint pathways.
- Include Next Links: shows the order of the reactions. From a node, an arrow indicates which node is the next step. This feature provides a timeline of the events in a pathway and could emphasize, for example, the linearity of a cascade.
- Include Pathways: includes green nodes (figure ??) which correspond to the names of the different pathways of the network.
- Include interactions: shows explicitly the reactions involved in the pathway (lower grey nodes in figure ??).

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Import BioPAX 3 Document from URL**  
A BioPAX 3 document can also be imported directly from a URL. The web address must be typed in the dialog window.

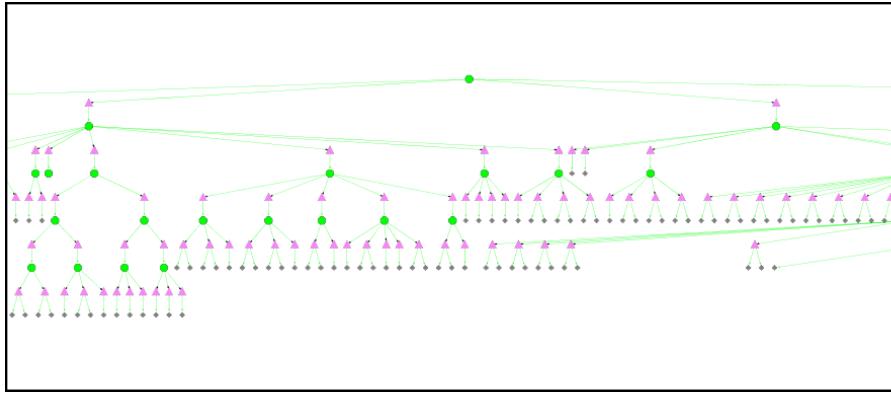


Figure 2: Apoptosis pathway hierarchical structure. Green nodes represent pathways, pink triangular nodes represent steps, and grey nodes represent reactions.

## 2.2 Import CellDesigner document

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Import CellDesigner Document from file**

The model can be drawn or downloaded[?] in CellDesigner (figure ??) and saved as M-Phase.xml.

The Import CellDesigner Document from file function imports a model from CellDesigner to Cytoscape. A dialog window opens and M-Phase.xml needs to be selected and imported (figure ??).

Figure ?? and ?? show the same model viewed by CellDesigner and Cytoscape respectively. The layout information from CellDesigner is imported automatically into Cytoscape.

In species notes in CellDesigner Attribute name:Value as HUGO:E2F1 (without blank) is converted in Cytoscape as the attribute HUGO with the value E2F1 for the specie.

## 2.3 Import CSML document

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Import CSML document**

BiNoM imports a CSML (Cell System Markup Language, csml.org)

## 2.4 Import influence network from AIN file

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Import AIN file**

This option proposes to automatically import an influence network in Cytoscape from a simple tab-delimited text file. We call the format of the text file AIN (Annotated

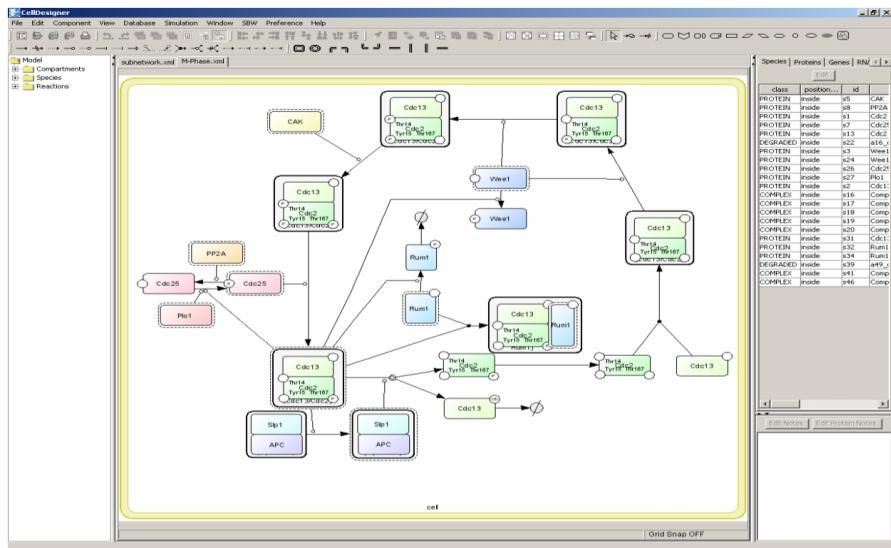


Figure 3: CellDesigner view of the cell division cycle model of fission yeast[?]

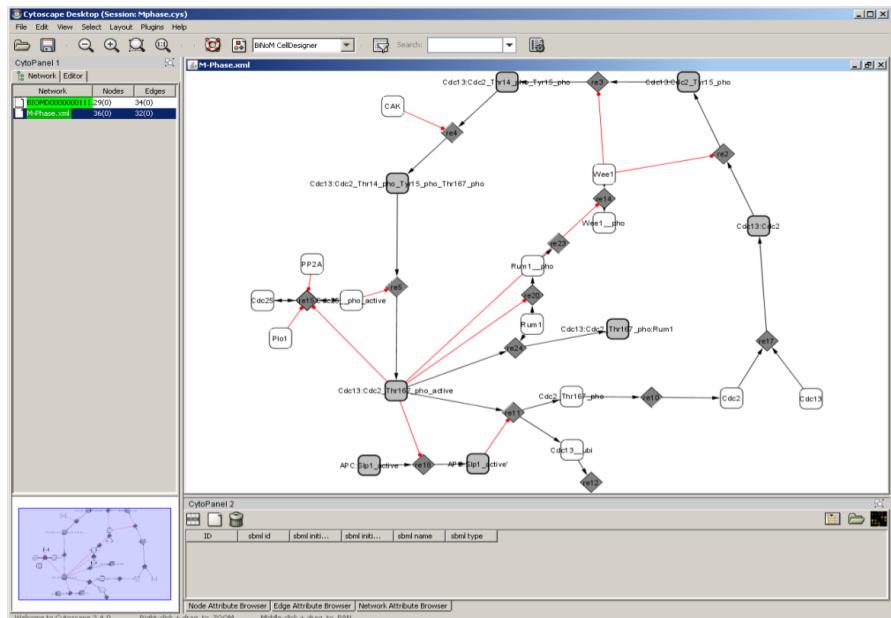


Figure 4: Cytoscape view of the cell division cycle model of fission yeast from a CellDesigner document

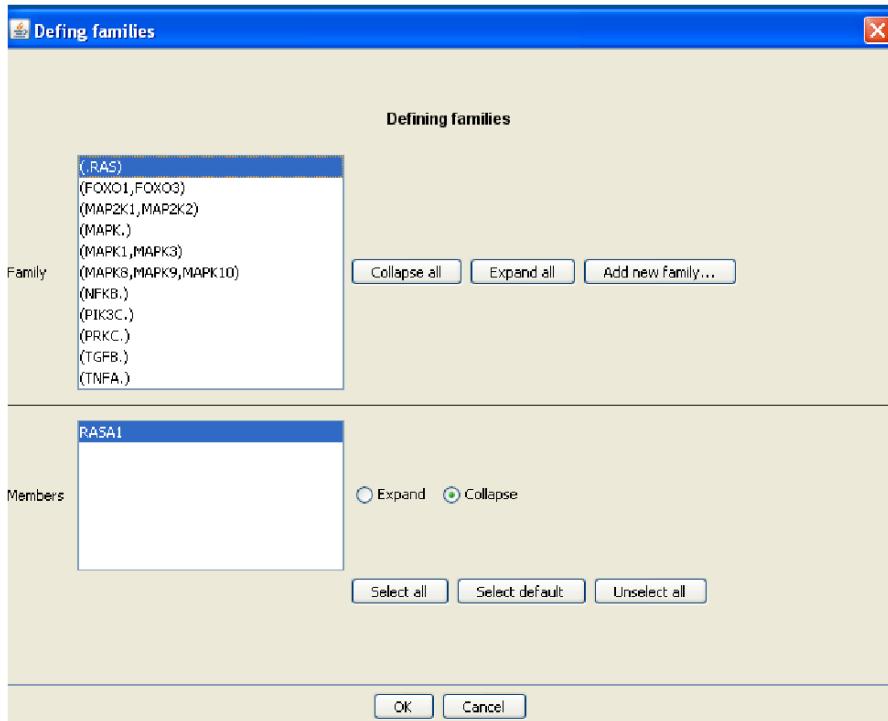


Figure 5: Dialog window for families management when importing the apoptosis network influence file (AIN format). BiNoM automatically detects gene families and proposes to either collapse or expand them.

Influence Network). Basically each row of the text file is encoding an interaction between two species, with a few optional descriptive fields. The precise format for each field is described in the appendices (section ??).

The AIN file of the apoptosis model ExamplApop.txt is imported. First, the user is asked to manage the families (groups of genes or proteins, see the appendices for a precise description): they can be expanded (replacing the family by all its members) or collapsed (replacing all family members by the name of the family). See figure ??.

Then a dialog window proposes to add constitutive reactions: influences that link proteins (or families) to their complexes and proteins (or families) to their phosphorylated state. See figure ??.

The imported network is synchronized with BioPAX format that includes the annotations of the AIN file. All this information can be accessed via BioPAX 3 property editor (see BioPAX Utils, section ??).

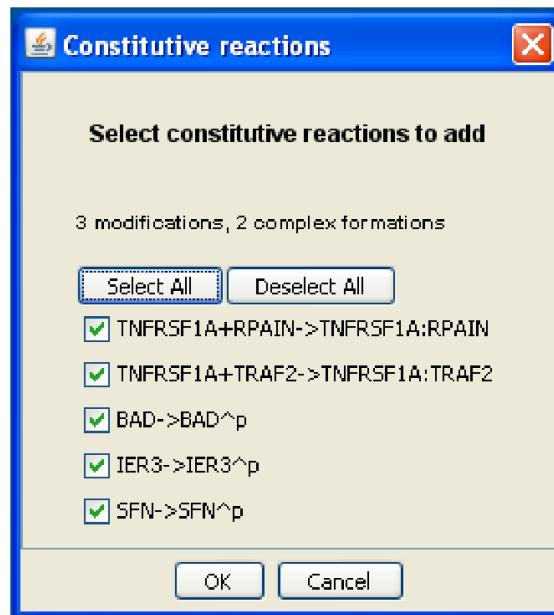


Figure 6: Dialog window for constitutive reactions when importing apoptosis influence file. BiNoM detects all possible constitutive reactions and proposes to add them.

## 2.5 Import reaction network from BiNoM reaction format file

This function is available in the experimental version of BiNoM. A network can be written as a list of reactions. The network can be a biochemical reaction network or an influence network following the notation below.

<p><b>Reactions</b></p> <ul style="list-style-type: none"> <li>-&gt; STATE_TRANSITION</li> <li>--&gt; KNOWN_TRANSITION_OMMITED</li> <li>-?&gt; UNNKNOWN_TRANSITION</li> <li>-+&gt; POSITIVE_INFLUENCE</li> <li>- &gt; NEGATIVE_INFLUENCE</li> <li>-/&gt; TRANSPORT</li> <li>-..&gt; TRANSCRIPTION</li> <li>-.&gt; TRANSLATION</li> <li>-:&gt; HETERODIMER_ASSOCIATION</li> <li>-=&gt; DISSOCIATION</li> </ul>	<p><b>Regulators</b></p> <ul style="list-style-type: none"> <li>- CATALYSIS</li> <li>- UNKNOWN_CATALYSIS</li> <li>-  INHIBITION</li> <li>-  UNKNOWN_INHIBITION</li> <li>-* MODULATION</li> <li>-) PHYSICAL_STIMULATION</li> </ul>
<b>Species</b>	
Component1 Modifname_Modiftype:Component2 Modifname_Modiftype@compartment	

Here we give an example of an influence network:

```
MOMP-->CYTOCHROME_C
APAF1-->CASP9
SMAC-|>XIAP
FADD-->CASP8
CASP8-->CASP3
CASP8-->BID
BAX-|>MOMP
CASP9-->CASP3
XIAP-|>CASP3
XIAP-|>CASP9
BID-->MOMP
```

The file must be saved in a txt format (with extension .txt). Here, the file is saved: apoptosis.txt

A network written with the BiNoM reaction format can then be imported in Cytoscape.

**Plugins⇒BiNoM 2.5⇒BiNoM I/O⇒Import reaction network from BiNoM reaction format file**

Choose apoptosis.txt

## 2.6 Export current network to BioPAX 3 or to CellDesigner

The Cytoscape networks can be exported in BioPAX and CellDesigner by:

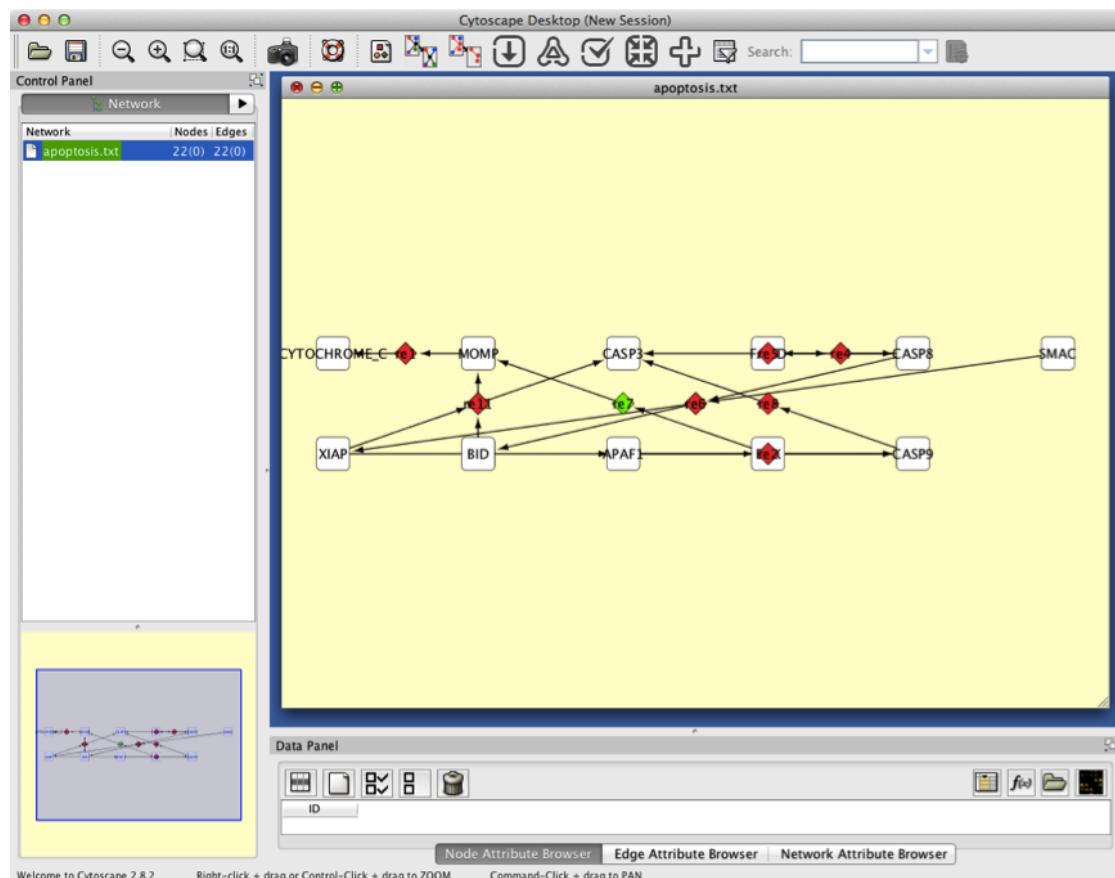


Figure 7: Network written with BiNoM format file and imported in Cytoscape. No layout is applied here.

- Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Export current network to BioPAX 3
- Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Export current network to CellDesigner

 The network to be exported should be associated to an existing CellDesigner or BioPAX file by using the functions:

- Associate BioPAX Source, see section ??.
- Associate CellDesigner Source, see ??.

BiNoM is able to convert CellDesigner to BioPAX, and BioPAX reaction network interface to pure SBML. BiNoM can also export only a part of a CellDesigner and BioPAX file, visible in the current Cytoscape network (interface). During the export operation, BiNoM can merge a part of associated BioPAX file with another part that has been saved already. BiNoM can modify the content of a BioPAX file.

 BiNoM is NOT able to create a CellDesigner file with all the graphical notations from a BioPAX file or from scratch, and it is also not able to modify the content of a CellDesigner file.

Here are a couple of typical scenarios where BiNoM export operations can be useful.

1. User imports a big BioPAX file as reaction network and using Cytoscape creates a new subnetwork from the global reaction graph. After he can export this subnetwork into a separate self-containing BioPAX file.
2. User imports the pathway structure of a big BioPAX file and selects only a few pathway or pathwayStep nodes he is interested in. After he can export a part of the BioPAX file necessary to define these pathways.
3. User imports a BioPAX file as reaction network, selects a subnetwork and exports it as pure SBML to be used for creation of a computational model of this subpart later.
4. User imports CellDesigner file, selects a subnetwork and exports it as a CellDesigner file: it can be useful for creating a CellDesigner image of a network module of a big reaction network.
5. User imports CellDesigner file, selects a subnetwork and exports it as a BioPAX file (some SBML-specific information such as parameters values will be lost).

The networks created as a result of the import operation are already associated to the corresponding BioPAX or CellDesigner files. However, if the XGMML file is saved and used in another Cytoscape session, or if a new network is created from the initial network with Cytoscape New menu then this association is lost.

To perform export operation, the network should be Re-associated to the corresponding file (from which it is originated) through Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Associate... operation. For huge BioPAX files the association might take some time for the first association, but once the file is loaded into memory cache, the following associations are almost instantaneous.

To understand better what BiNoM can do or can not, read the sections ??, ?? and ?? about the BiNoM data model.

## 2.7 Create CellDesigner file from current network

This function is available in the experimental version of BiNoM.

**Plugins⇒BiNoM 2.5⇒BiNoM I/O⇒Create CellDesigner file from current network**

The file needs to be saved with the extension .xml.

This function is different than:

**Plugins⇒BiNoM 2.5⇒BiNoM I/O⇒Export current network to CellDesigner**

With this export function, the user needs to associate the network to an existing CellDesigner file. It is useful when creating a subnetwork from an existing CellDesigner network, for instance.

Here we save our network with the name: apoptosis.xml.

Open the saved file into CellDesigner. Apply the orthogonal layout or re-arrange the network as you wish.

The network exported in CellDesigner can also be a network imported from a BioPAX file. This function allows the visualization of a BioPAX file in CellDesigner. For that, you need to import a BioPAX file using the function:

**Plugins⇒BiNoM 2.5⇒BiNoM I/O⇒Import reaction network from BioPAX**

- Importing any type of BioPAX file

Here we choose exampleBIOPAX.owl.

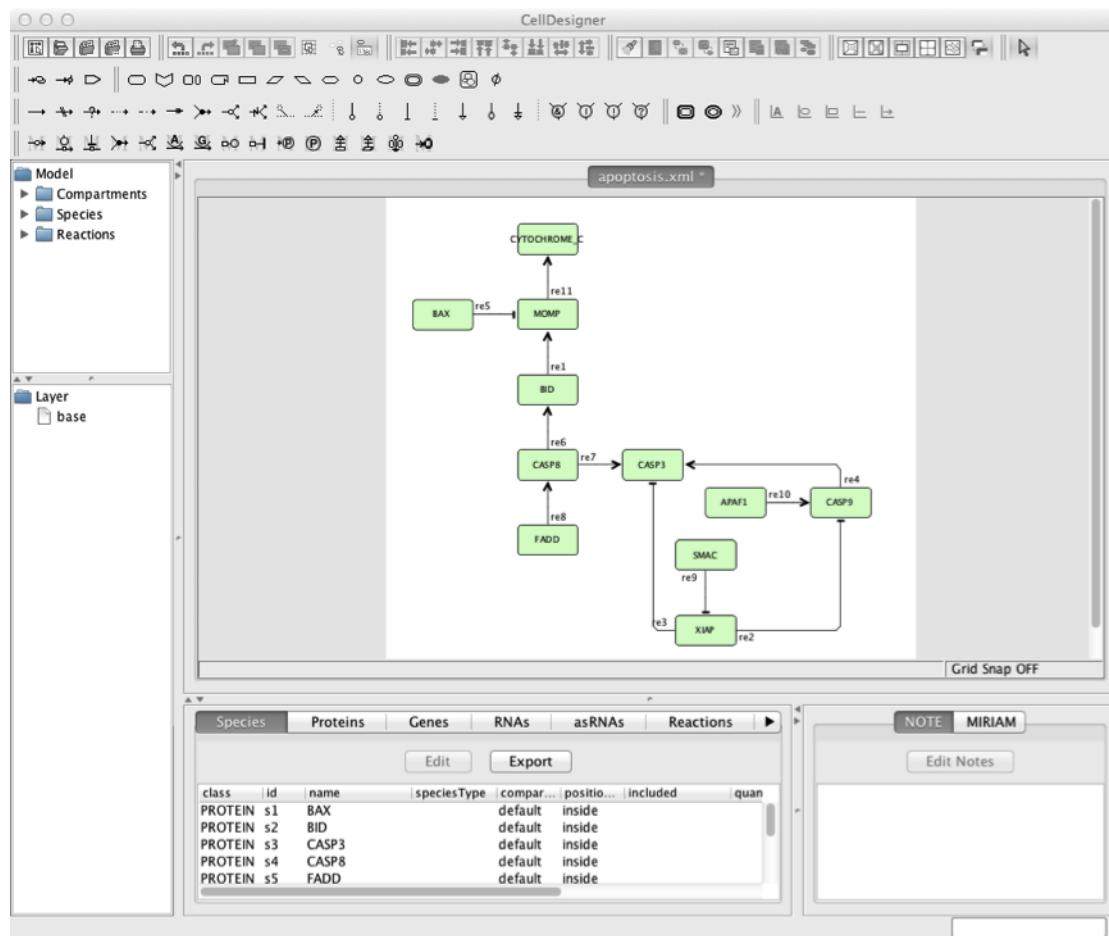
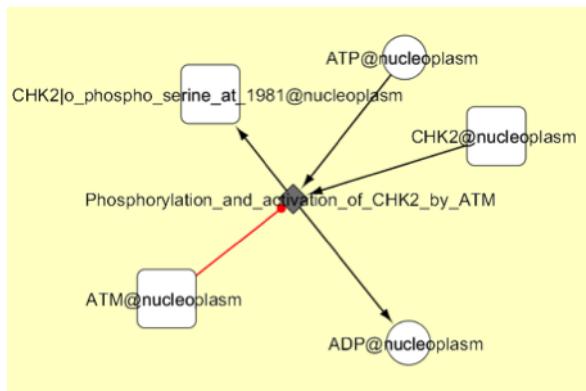


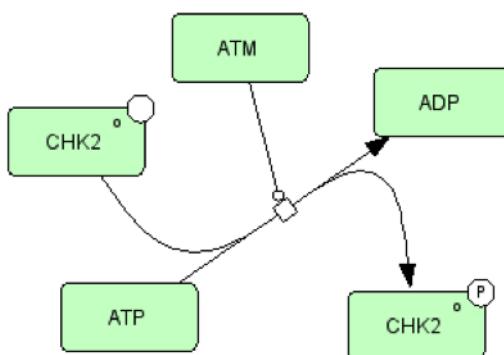
Figure 8: Network written with BiNoM format file and imported in CellDesigner. Orthogonal layout is applied here.



Then, export the current network to CellDesigner using the function:  
**Plugins⇒BiNoM 2.5⇒BiNoM I/O⇒Create CellDesigner file from current network**

Save the file as: example.BIOPAX.xml

Open the file in CellDesigner:



- Importing a BioPAX file from WikiPathways  
 Here are the steps:

1. Download the pathway *SenescenceandAutophagy(Homosapiens)* and save the file *WP61560842RN* from WikiPathways website;
2. Import network in Cytoscape with the function: **Plugins⇒BiNoM 2.5⇒BiNoM I/O⇒Import reaction network from BioPAX**
3. Create a CellDesigner file from this network: **Plugins⇒BiNoM 2.5⇒BiNoM I/O⇒Create CellDesigner file from current network**  
 And save it as: *WP61560842.xml*.
4. Launch CellDesigner and open the file: *WP61560842.xml*
5. Apply a layout (here organic).

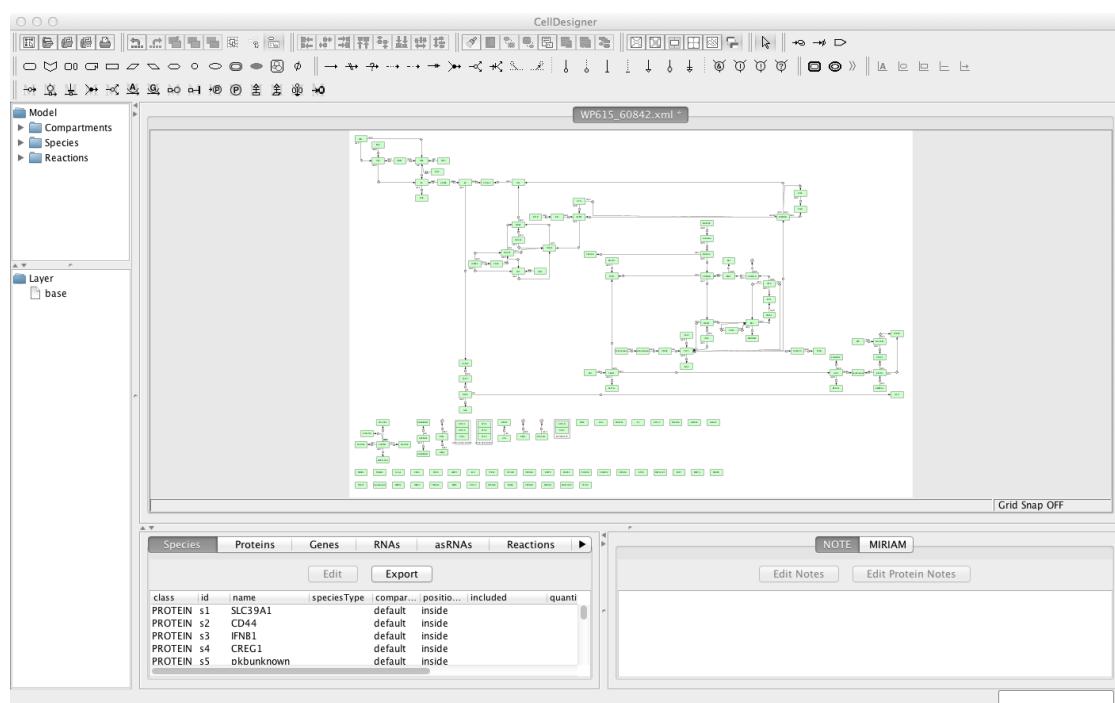


Figure 9: Network downloaded in WikiPathways and imported in CellDesigner. Orthogonal layout is applied here.

## 2.8 Export current network to SBML

Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Export current network to SBML  
Export the current network to pure SBML level 2.

## 2.9 Export network to BiNoM reaction format

Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Export network to BiNoM reaction format

This function allows to translate a network into a list of reactions. The file is saved as a text file.

As an example, we imported the file M-Phase.xml and exported the network to BiNoM reaction format file. The file is saved as a txt file (MPhasebrff.txt) as follows:

```
(APC' :Slp1')||active'|active
(APC:Slp1)||active
Cdc13:Cdc2|Thr167_phos:Rum1
Rum1|pho
Rum1
Cdc13|ubi
(Cdc13:Cdc2|Thr167_phos)||active
Cdc13:Cdc2|Thr14_phos|Tyr15_phos|Thr167_phos
Cdc13:Cdc2|Thr14_phos|Tyr15_phos
Cdc13:Cdc2|Tyr15_phos
Cdc13:Cdc2
Cdc13
Plo1
Cdc25|pho|active
Wee1|pho
Wee1
Cdc2|Thr167_phos
Cdc25
Cdc2
PP2A
CAK

Rum1+(Cdc13:Cdc2|Thr167_phos)||active->Cdc13:Cdc2|Thr167_phos:Rum1
Rum1|pho->null
Rum1-(Cdc13:Cdc2|Thr167_phos)||active->Rum1|pho
(APC:Slp1)||active-(Cdc13:Cdc2|Thr167_phos)||active->(APC' :Slp1')||active'|active
Cdc13+Cdc2->Cdc13:Cdc2
Cdc25|pho|active-PP2A->Cdc25
Cdc25-(Cdc13:Cdc2|Thr167_phos)||active-Plo1->Cdc25|pho|active
```

```

Wee1-(Cdc13:Cdc2|Thr167_phos)||active->Wee1|pho
Cdc13|ubi->null
(Cdc13:Cdc2|Thr167_phos)||active-(APC':Slp1')||active'|active=>Cdc13|ubi+Cdc2|Thr167_phos
Cdc2|Thr167_phos->Cdc2
Cdc13:Cdc2|Thr14_phos|Tyr15_phos|Thr167_phos-Cdc25|phos|active->(Cdc13:Cdc2|Thr167_phos)||active'
Cdc13:Cdc2|Thr14_phos|Tyr15_phos-CAK->Cdc13:Cdc2|Thr14_phos|Tyr15_phos|Thr167_phos
Cdc13:Cdc2|Tyr15_phos-Wee1->Cdc13:Cdc2|Thr14_phos|Tyr15_phos
Cdc13:Cdc2-Wee1->Cdc13:Cdc2|Tyr15_phos

```

Note that BiNoM lists first all the species and then the reactions.

We propose three scenarios that use BiNoM reaction format file.

- First scenario: create a CellDesigner file from a textual model

1. Write network in text format:

```

A => B
B+C=>D+E
...

```

2. Import reaction network in Cytoscape using BiNoM
3. Create CellDesigner file from current network
4. Open CellDesigner and import the newly saved file

- Second scenario: create a CellDesigner file from BioPAX file

1. Import a BioPAX file in Cytoscape
2. Create CellDesigner file from current network
3. Open CellDesigner and import the newly saved file

## 2.10 Associate BioPAX 3 Source

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Associate BioPAX 3 Source**

Associate a BioPAX 3 Source to allow exportation in BioPAX 3 as explained in section ??

## 2.11 Save whole associated BioPAX 3 as

When the content of the BioPAX file is modified (through the BioPAX property editor, see section ??), it can be saved as a whole (not only the visible part) by

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Save whole associated BioPAX 3 as**

Otherwise, all modifications made in the different interfaces are lost. Changes are visible but only recorded permanently when the document is saved to a file.

## 2.12 Associate CellDesigner Source

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Associate BioPAX 3 Source**

Associate a CellDesigner Source to allow exportation in CellDesigner as explained in section ??

## 2.13 List all reactions

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒List all reactions**

Display list of reactions, can be copied by control+A then control+C.

## 2.14 List all nodes

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒List all nodes**

Display list of nodes, can be copied by control+A then control+C.

## 2.15 Color CellDesigner proteins

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Color CellDesigner proteins**

Cytoscape allows coloring nodes according to values of attributes (for example expression data) by the powerful possibilities of VizMapper. The export to CellDesigner keeps the colors. This process can be used to color species in CellDesigner. The function Color CellDesigner proteins allows to color proteins in CellDesigner which describe the components of complexes.

The gene expression file is based on Hugo names, data in columns (first line title and tabulation as column separator):

Hugo names<Tab>expression level 1<Tab>expression level 2...

Open dialog box Color CellDesigner proteins, input CellDesigner file name and gene expression file, click on ok. BiNoM generate a file \*.conv where Hugo names are converted in protein names (links by annotation in CellDesigner, check if correct) and a CellDesigner file by column. When there are several Hugo name the highest is kept.

Figure ?? shows the aspect of colored proteins inside complexes.

## 2.16 Modify CellDesigner notes

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Color CellDesigner proteins**

Modify in Cytoscape the notes of CellDesigner file when exporting.

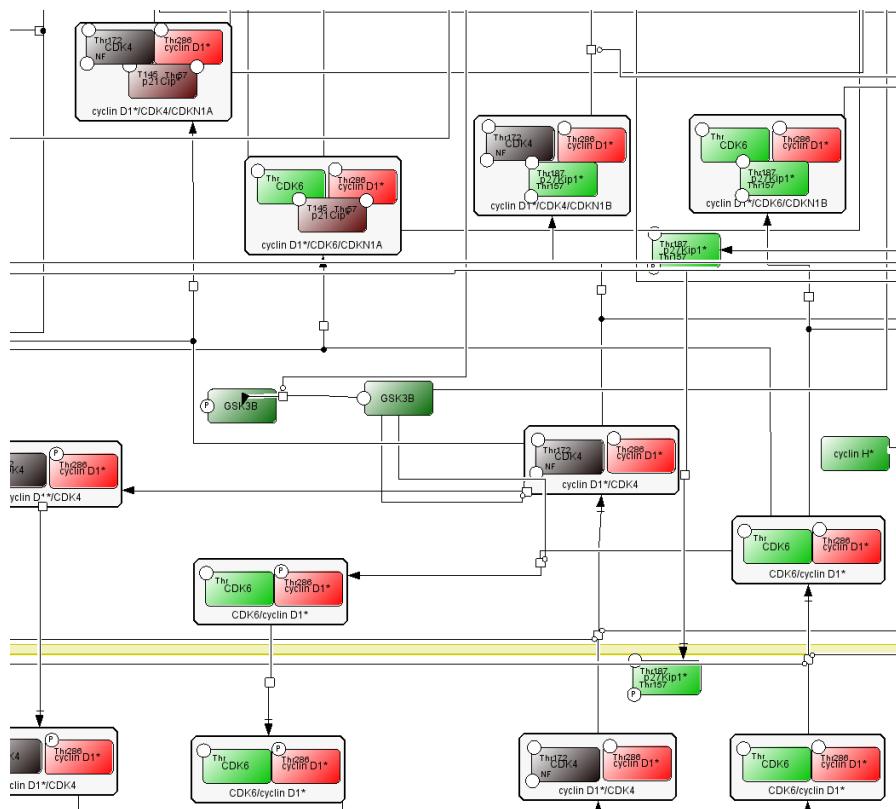


Figure 10: CellDesigner view of an extract from Rb-E2F[?] pathway colored by fictitious expression data

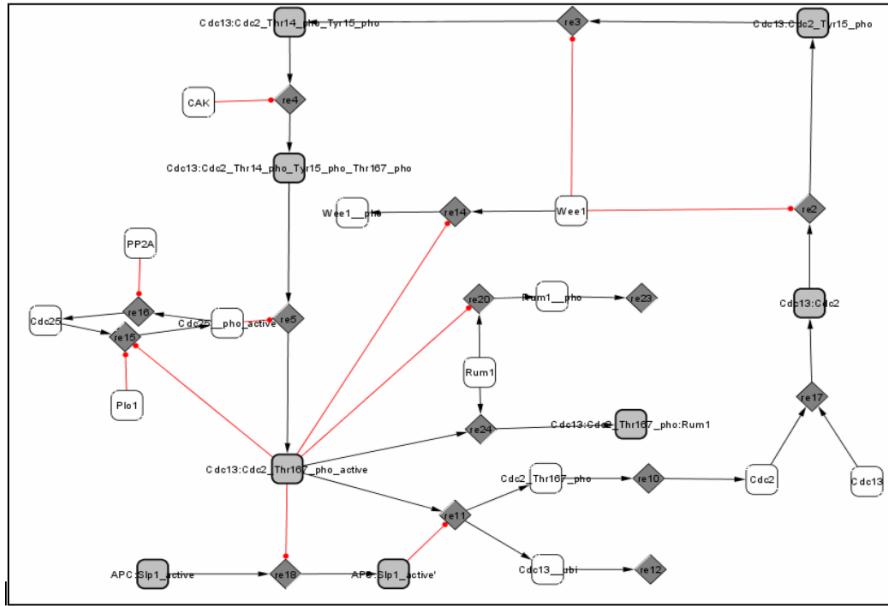


Figure 11: Cytoscape view of the M-Phase network

### 3 BiNoM Analysis

We illustrate, here, the different functions of BiNoM related to the structural analysis, using the modified version of the Novak et al. model, M-Phase.xml as an example (figure ??).

From the menu Plugins⇒BiNoM 2.1⇒ BiNoM analysis, we review all the functions one by one.

#### 3.1 Get connected components

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Get connected components**

This command dissociates the unconnected subparts of the network. In our case, since the network is already completely connected, the one obtained when choosing this function is the same as the initial one (called M-Phase.xml\_cc1).

#### 3.2 Get strongly connected components

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Get strongly connected components**

Based on Tarjans algorithm[?], the strongly connected components are isolated. In simple words, the obtained network, M-Phase.xml\_scc1(figure ??), insures that there exists a path from one node to another and deletes the components which do not respond to this requirement.

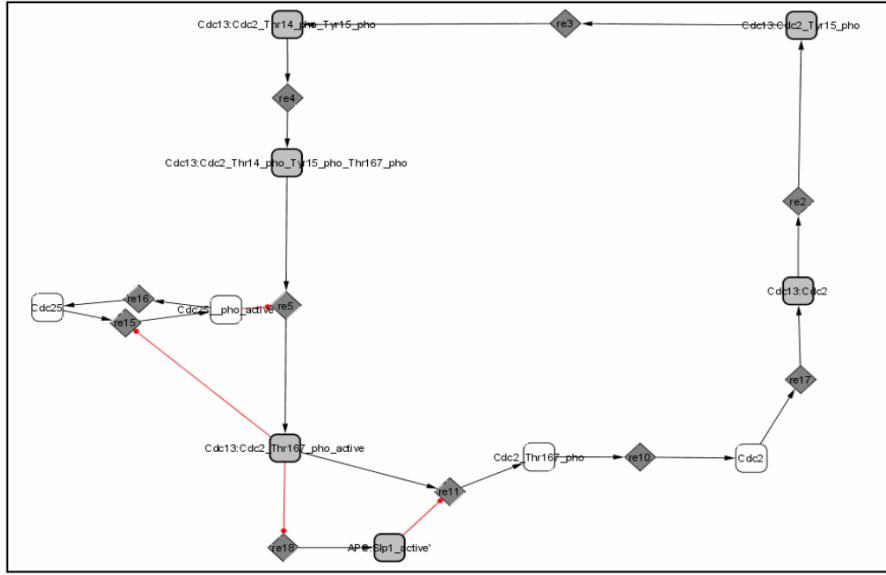


Figure 12: Strongly Connected Component of M-Phase network

### 3.3 Prune Graph

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Prune graph**

Pruning the graph is equivalent to separating the network into three parts(figure ??: what comes in (M-Phase.xml.in), what goes out (M-Phase.xml.out) and the central cyclic part (M-Phase.xml.scc)).

This decomposition corresponds to the idea of the bow-tie structure developed by Broder and colleagues[?]. In our example, the central cyclic part is the same as figure ??, the strongly connected component. In other cases, it can be composed from several strongly connected components, connected or disconnected.

The Prune graph operation decomposes the current network into three parts: IN, OUT and SCC (the later can contain several strongly connected components).

### 3.4 Get Material Components

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Get material components**

This function uses node name semantics to isolate sub-networks in which each protein takes part. In our example(figure ??), seven sub-networks are created: M-Phase.xml\_Cdc13, M-Phase.xml\_Cdc2, M-Phase.xml\_Rum1, M-Phase.xml(APC, M-Phase.xml\_Slp1, M-Phase.xml\_Cdc25 and M-Phase.xml\_Wee1. Some major overlaps between sub-networks are expected, as it is the case for Cdc2 and Cdc13 which form a complex.

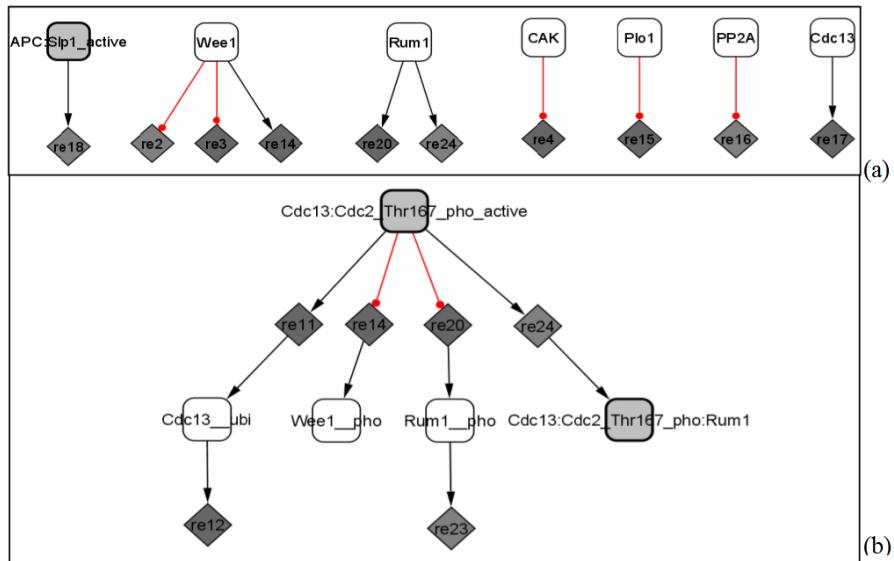


Figure 13: Prune the graph. (a) Incoming flux: molecules involved in the IN part of the network, and (b) Outgoing flux: molecules involved in the OUT part of the network.

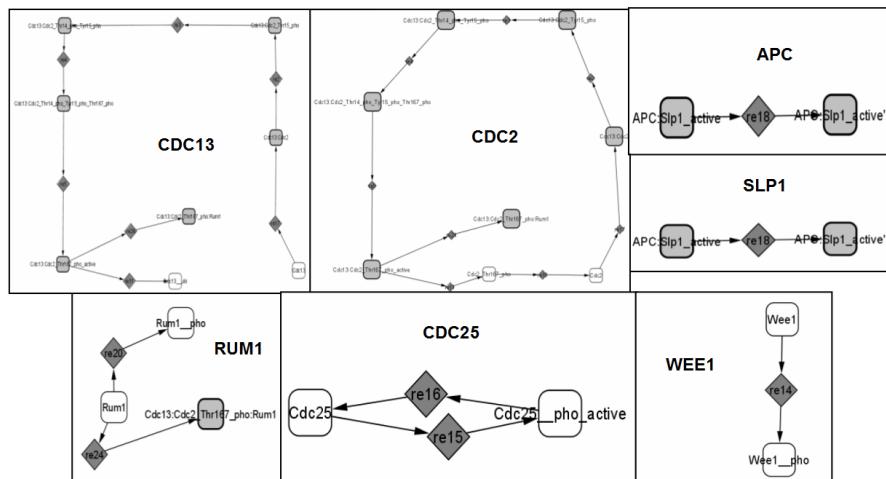


Figure 14: Material Components

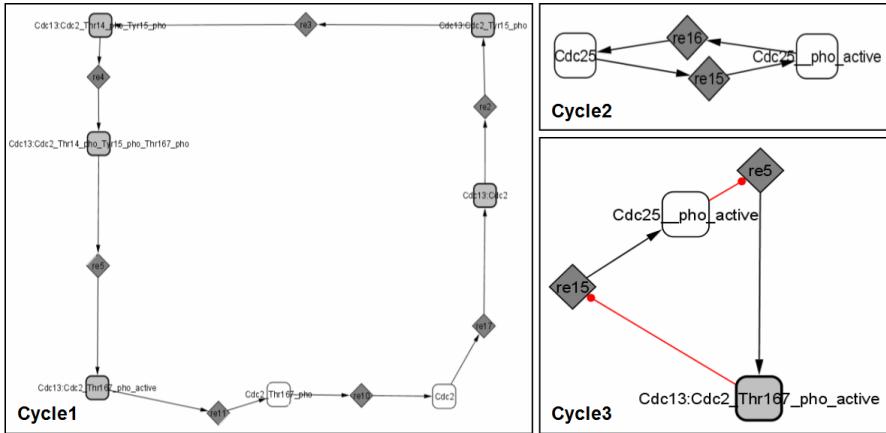


Figure 15: Minimal cycle decomposition of the M-Phase network. Cycle 1 includes CDC2 and CDC13 proteins, Cycle 2 CDC25 and Cycle 3 shows the feedback existing between CDC13/CDC2 and CDC25.

### 3.5 Get Cycle Decomposition

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Get cycle decomposition**

This command decomposes the network into relevant directed cycles[?], using a modification of the Vismaras algorithm[?]. Often, this feature gives information about the life cycle of a protein or a complex, about the feedbacks of the studied network, etc(figure ??). Note that the union of all the cycles corresponds to the strongly connected component figure ??.



This operation can produce enormous number of cycles! Therefore it is rather suitable for analysis of small to moderate size networks. For a big network, one can start to understand the cyclic network structure by eliminating first the network hubs, which are contained in many network cycles. After that, the local, relatively short, cycles can be represented as meta-nodes (modules) and the analysis for cycles can be repeated.

### 3.6 Path Analysis

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Path analysis**

In a network, it can become handy to find out if there exists a path (or paths) from one species to another, or to verify that a protein or a protein complex is reachable from a starting molecule(figure ??). Provided (an) initial source and target protein(s) that are selected first on the graph then in the dialog window, the command Path analysis can find: the shortest paths, the optimal and suboptimal shortest paths, or all the non-intersecting paths (does not include inner loops), using a finite number of intermediary nodes (use finite breadth search radius), for either directed or undirected paths (figure ??).

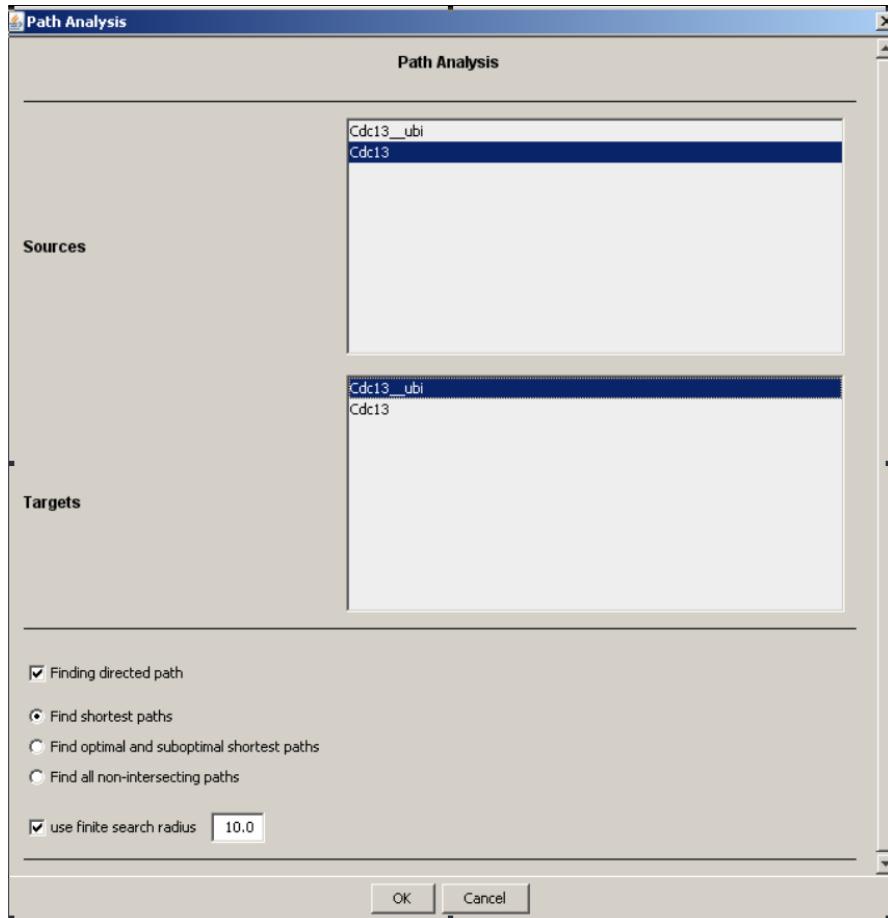


Figure 16: BiNoM Path Analysis: Pop-up window in which the source(s) and the target(s) need to be specified along with the type of paths (shortest, optimal shortest or all paths).



In big networks the number of paths can be exponential! It is recommended to find the shortest path first, take its length and increment gradually the breadth search radius starting from this value to find the second shortest, third shortest, etc., paths.

### 3.7 Extract subnetwork

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Extract subnetwork**

Extract a subnetwork from selected nodes of a network with various options (figure ??).

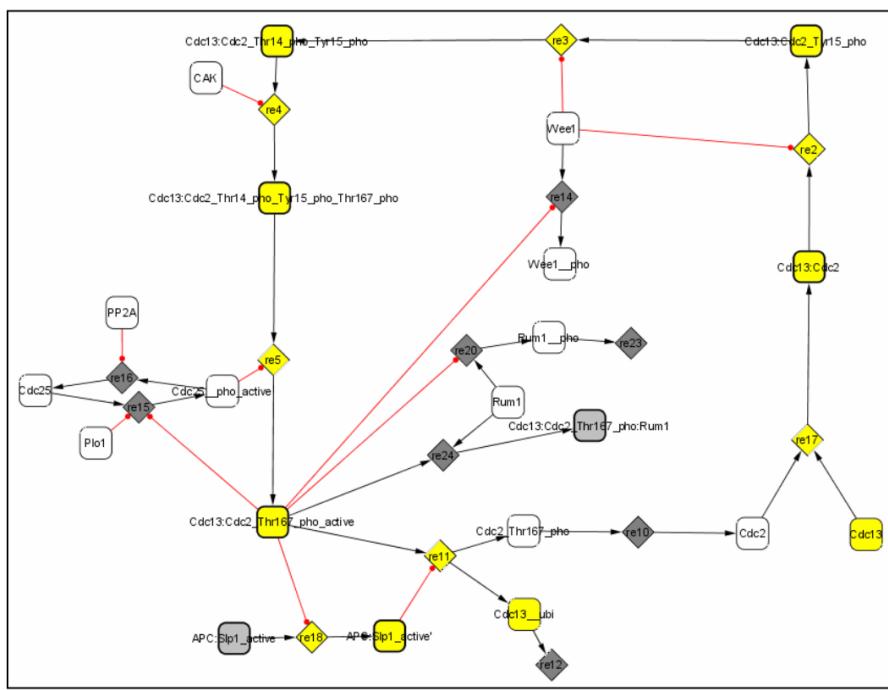


Figure 17: Path Analysis: All the paths leading from one molecular species (Cdc13) to another (Cdc13\_ubi, ubiquitinated form of Cdc13) are highlighted in yellow.

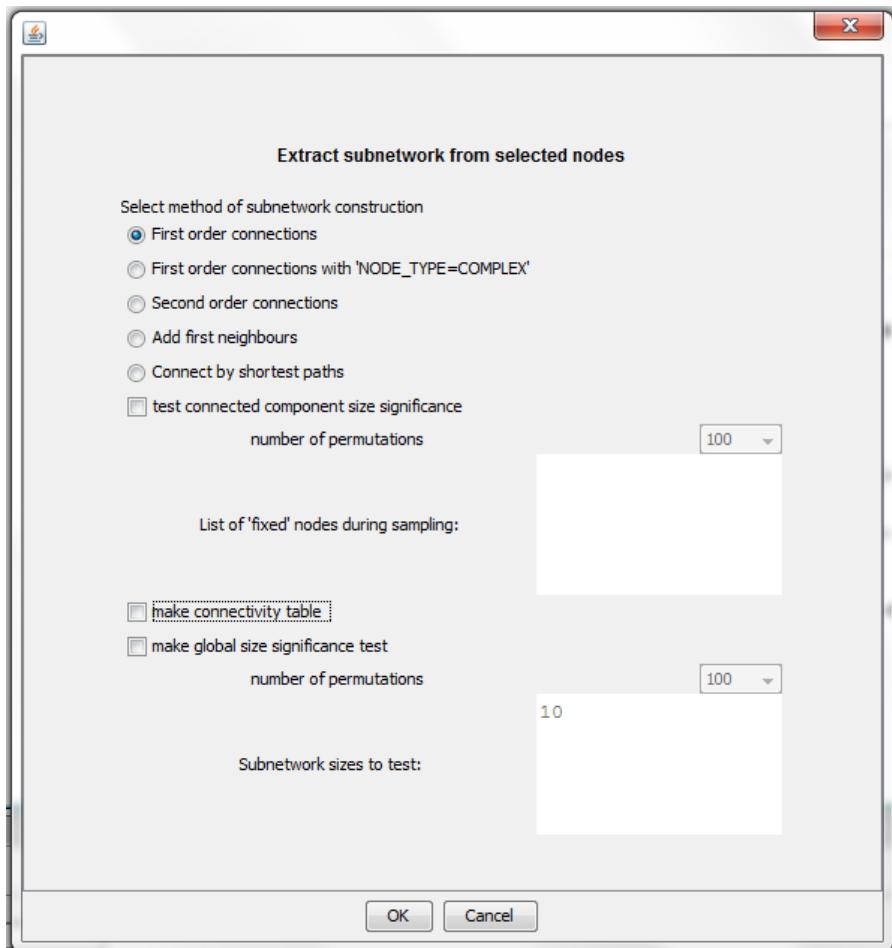


Figure 18: Dialog of the “Extract subnetwork” function showing the different options available.

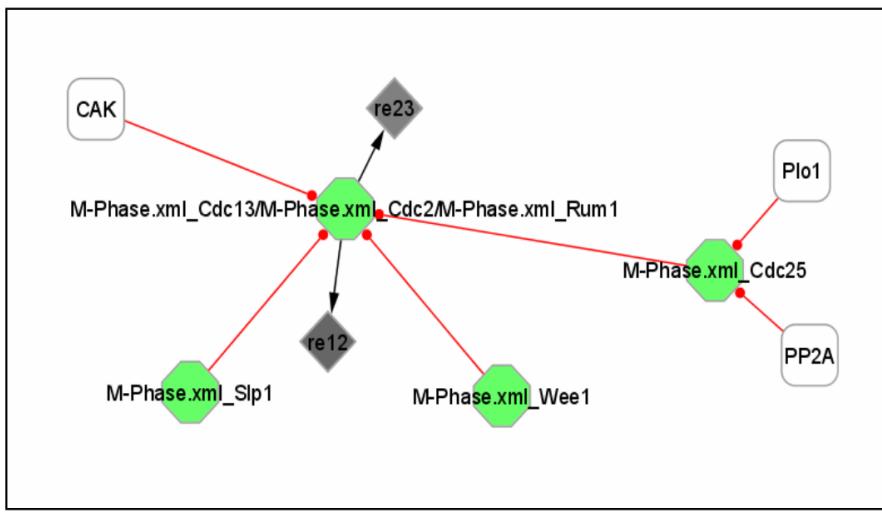


Figure 19: Clusters of modules. The obtained diagram is a compact modular view of the M-Phase network using the material decomposition and material components clustering

**3.8 Calc centrality, Inbetweenness undirected, Inbetweenness directed**  
**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Calc centrality⇒Inbetweenness undirected Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Calc centrality⇒Inbetweenness directed** Display centrality of nodes in cases undirected and directed.

### 3.9 Cluster Networks

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Cluster networks**

This command lumps together the modules that share a certain proportion of nodes. At a first glance, it can easily be concluded from Figure ?? (middle panel) that, for example, the modules M-Phase.xml\_Cdc13 and M-Phase.xml\_Cdc2 share a lot of proteins or protein complexes. Therefore, we can assume that these two modules will collapse into one big module. To determine the clusters, the intersection threshold can be set (from 0 to 100% intersecting components). For a 30% intersection threshold, Figure ?? is obtained. Four clusters of modules were proposed and linked.

An alternative modular view has been obtained using the cycle decomposition instead of the material decomposition. The cycles are presented in Figure ???. They are obtained by clustering the three cycles into two (cycle 1 + cycle2/cycle3) and organized into a modular view (Figure ??).

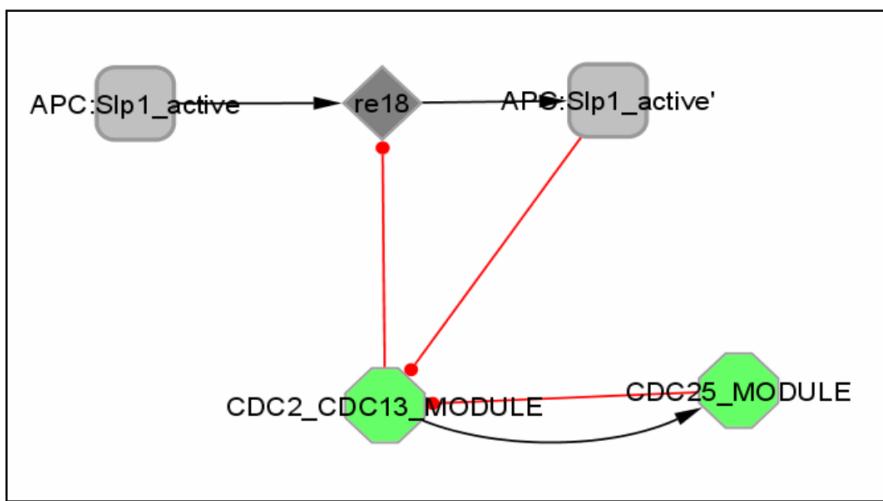


Figure 20: Clusters of modules. The obtained diagram is a compact modular view of the M-Phase network using the relevant cycle decomposition and cycle clustering

### 3.10 Mono-molecular react.to edges

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Mono-molecular react. to edges**

This function transforms monomolecular reaction nodes (i.e. reactions with one reactant and one product) into influence edges. Thus, monomolecular (linear) reactions are represented as edges and the reaction graph is not bi-partite anymore. When the reaction nodes have the type of influence specified (through the Cytoscape EFFECT attribute), the graph is transformed automatically into an influence graph (see Figure ??: upper panel: BioPAX network, lower panel, the equivalent influence network). Non-linear non-monomolecular reactions (such as complex assemblies) are not transformed and remain to be represented as network nodes.

### 3.11 Linearize network

The “linearization” replaces explicit reactions of complex formation by direct influences between constituents and complexes (in the case of an influence network imported from an AIN file).

To illustrate this function, import the AIN file *cell\_cycle\_AIN.txt*.

**Plugins⇒BiNoM 2.1⇒BiNoM I/O⇒Import influence network from AIN file...**

Click two times 'OK' for the specification of the families. A new network is created. Then, apply the linearization functions **Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Linearize network**

A new 'linearized' network is created (figure ??).

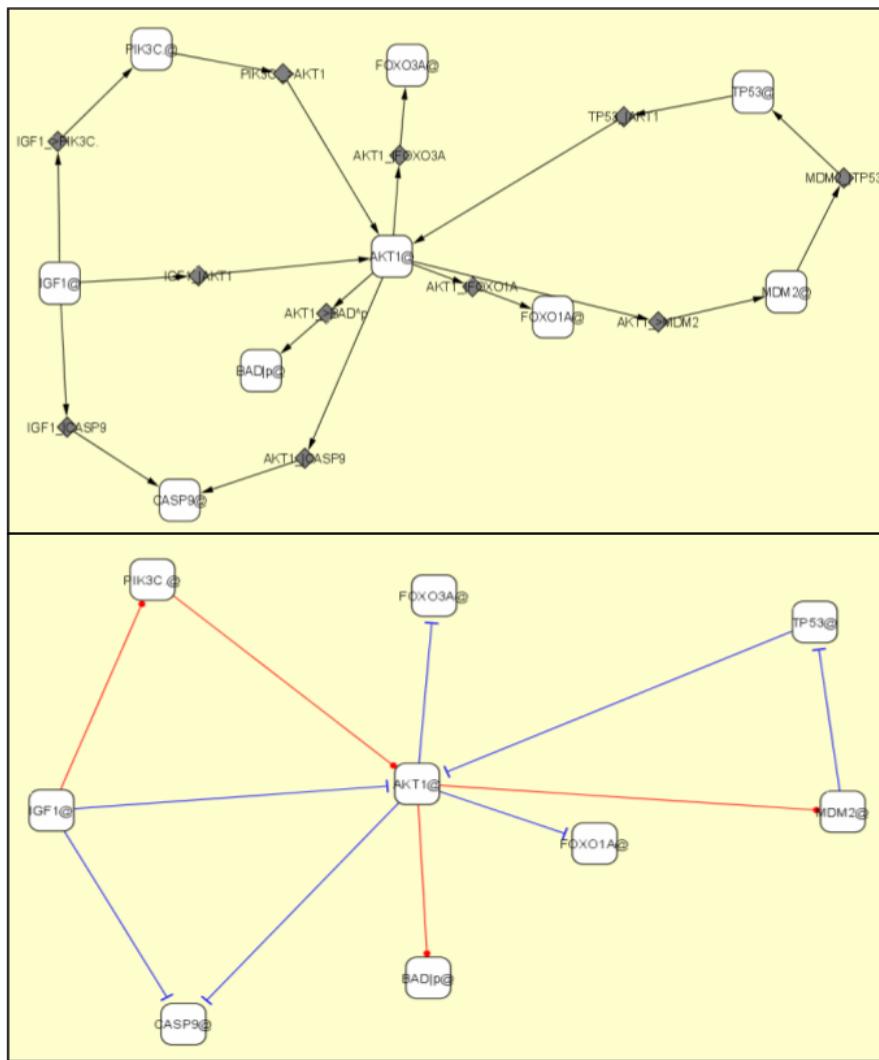


Figure 21: From a BioPAX network (upper panel) to an influence graph (lower panel).

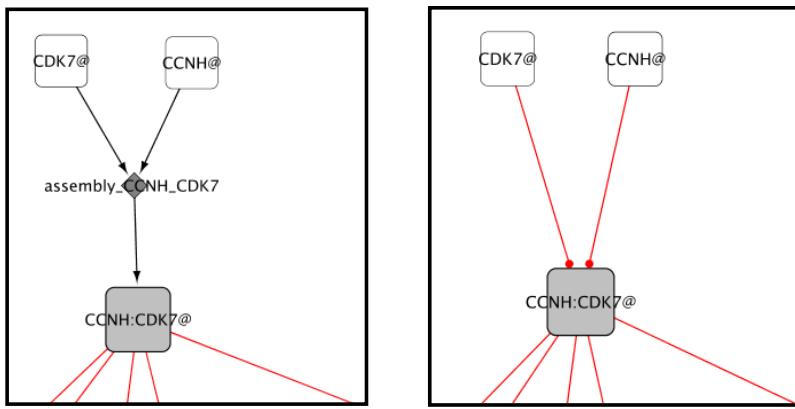


Figure 22: Example of explicit reaction of complex formation (left panel) that can be replaced by direct connections (right panel) using the “linearization” function of BiNoM.

### 3.12 Exclude intermediate nodes

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Exclude intermediate nodes**

This function opens a dialog where nodes to be excluded can be selected (figure ??). It creates a network without the selected nodes and reconnects edges.

### 3.13 Extract Reaction Network

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Extract reaction networks**

This function cleans up the diagram to only keep the reaction network. Only nodes with XXXX\_reaction and XXXX\_species attributes (where XXXX stands for any word) are kept as a result of this operation. For example, it helps to clean the reaction network interface from the result of querying BioPAX index (which contains many other node types such as entities and publications).

### 3.14 Path Influence Quantification analysis

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Path Influence Quantification analysis**

This function calculates a score to quantify the effect of experimental data onto one or more target nodes for a given network architecture, named the PIQuant score (Pathway Influence Quantification). The target node can be a gene or a phenotype of interest (such as cell cycle or apoptosis). We define as annotated nodes any node of the network for which we have experimental data available. The experimental data can be for instance an mRNA expression value (ratio disease/normal). We also assume that we have determined all the paths from the annotated nodes to the target nodes using an appropriate algorithm (such as Dijkstra’s shortest paths algorithm). The PIQuant score

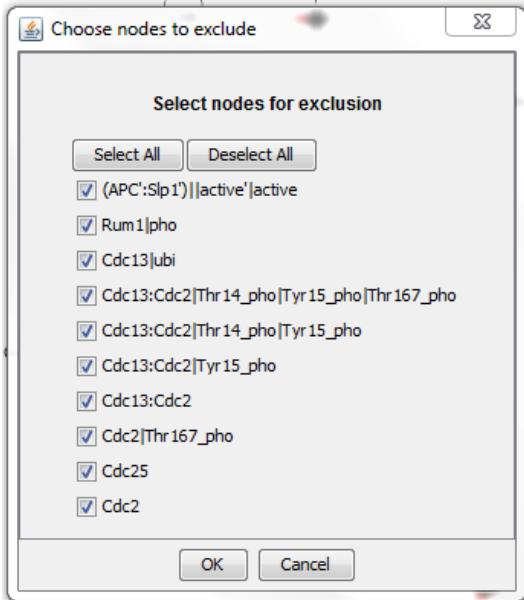


Figure 23: Dialog to select nodes to be excluded in the created network.

is then defined as:

$$PIQuant_{Score} = \sum_{k=1}^q \alpha_k \sigma_k \frac{1}{\lambda_k}$$

A path  $k \in \{1, \dots, q\}$  is defined as a the sequence of consecutive connected nodes between an annotated node and a target node (without repetition of any node or edge).

The annotation  $\alpha_k$  of the path  $k$  is defined as the annotation (real value representing the experimental data) of the first node in the path. We define the sign  $\sigma_k$  of the path  $k$  as the product of the signs of every edge of the path and finally the length  $\lambda_k$  of the path  $k$  as the number of edges in the path. We hypothesize that the longer the path is, the lesser the global influence will be on the target node.

**!** The Cytoscape edge attribute which encodes the influence corresponding to the sign (+1 or -1) of the edge is "EFFECT" which can take 2 values EFFECT:activation and EFFECT:inhibition.

In the case of the network presented in figure ??a, let us consider Ac the annotated node and Ph the target node and consider only the two paths defined in the figure ??b and figure ??c. Given that the node Ac is annotated by the value 2.0, that the first path has a length equal to 3, and that the second path has a length equal to 5, we can calculate the PIQuant score of the node Ac to the node Ph as:

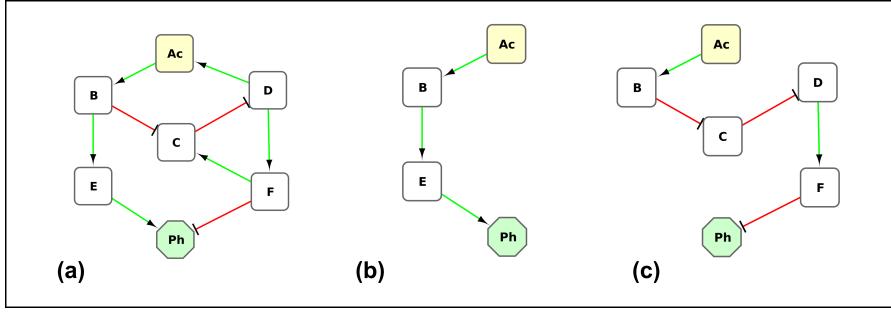


Figure 24: A simple influence network. The network is composed of seven nodes and nine edges (a). The two paths (b,c) extracted from this network start from the source node **Ac** and end at the target node **Ph** (a phenotype of interest). The nodes **Ac** and **D** are annotated using experimental data, and have the values 2.0 and -1.0 respectively.

$$PIQuant_{Score} = 2 \cdot 1 \cdot \frac{1}{3} + 2 \cdot (-1) \cdot \frac{1}{5} = 0.27$$

Sometimes a path has one or more intermediate nodes that are annotated nodes (i.e. for which we have experimental data values), in addition to the source node. In that case, the intermediate node is *consistent* if the sign of its annotation is the same as the sign of the first node in the path multiplied by the sign of the path from the annotated node to the intermediate node. If signs are opposite, the node is *inconsistent*. A path is *consistent* if each annotated intermediate nodes is consistent. A path is *inconsistent* if at least one intermediate node is *inconsistent*. In other words, a path is inconsistent when the sign of the experimental data value for a node is opposite to the sign of the path.

For example, according to the network represented in figure ??c, there is an influence from Ac to Ph that goes through D . But if this path is functional, the annotation of node D should have the same sign as the annotation for node Ac, because Ac activates D indirectly (i.e. the path from Ac to D has a positive sign). In this case, the sign of D annotation (the experimental value) is opposite to the sign of Ac annotation, and therefore, the path from Ac to Ph that goes through D is inconsistent.

Practically, we offer the option to keep or not the inconsistent paths for the calculation of the PIQuant score, depending on how the user wants to analyze the calculations. An inconsistent path could indicate that the path is not complete, or could also indicate that the path is correct but not active under the precise conditions in which the experimental data was generated, corresponding to a different context.

In the case mentionned above in figure ??c, if only consistent paths are kept, then the PIQuant score of Ac to Ph becomes:

$$PIQuant_{Score} = 2 \cdot 1 \cdot \frac{1}{3} = 0.67$$

This score is higher than the value previously obtained with all the paths, and can

be interpreted in this case as a higher activation of the phenotype.

In order to calculate the PIQuant score, an influence network with annotated nodes should be created, for example using the AIN file format (see the AIN file format description in the appendix).

Once the network is created in Cytoscape, then we can call the BiNoM function to calculate the PIQuant score.

### **Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Path Influence Quantification analysis**

This function has two dialogs, one for choosing the annotated nodes, targets nodes and path search algorithm, and one to display the results of the calculation.

Step 1 see figure ??:

- Select the attribute name corresponding to the annotation and click update list, only active nodes are displayed in box.
- Select target nodes.
- Choose one algorithm for path searching, see glossary in section ?? for details on the different algorithms.
- Click OK.

The step 2 dialog (the results of the PIQuant score calculations) is displayed on figure ??.

All the paths and the corresponding PIQuant scores are displayed in three interactive lists on the upper part of the window. The user can select the target nodes, annotated nodes and the paths. Individual PIQuant score are displayed for each path, while a sum of the individual PIQuant scores is displayed below the lists.

Different options are available, such as using sign algebra, selecting all paths, filter out inconsistent nodes, filter out inconsistent paths, filter inactive paths and get Paths activities.

It is also possible to test the significance of a global PIQuant score, by reshuffling the annotations between the nodes and estimating a p-value from the distribution.

At last, the user can get a text report, detailing the global and individual PIQuant scores.

### **3.15 Create neighborhood sets file**

**Plugins⇒BiNoM 2.1⇒BiNoM Analysis⇒Create neighborhood sets file**  
This function creates a file \*.gmt (a text file where nodes are separated by <Tab>) containing the neighbors of selected nodes according to option of the dialog(figure ??)

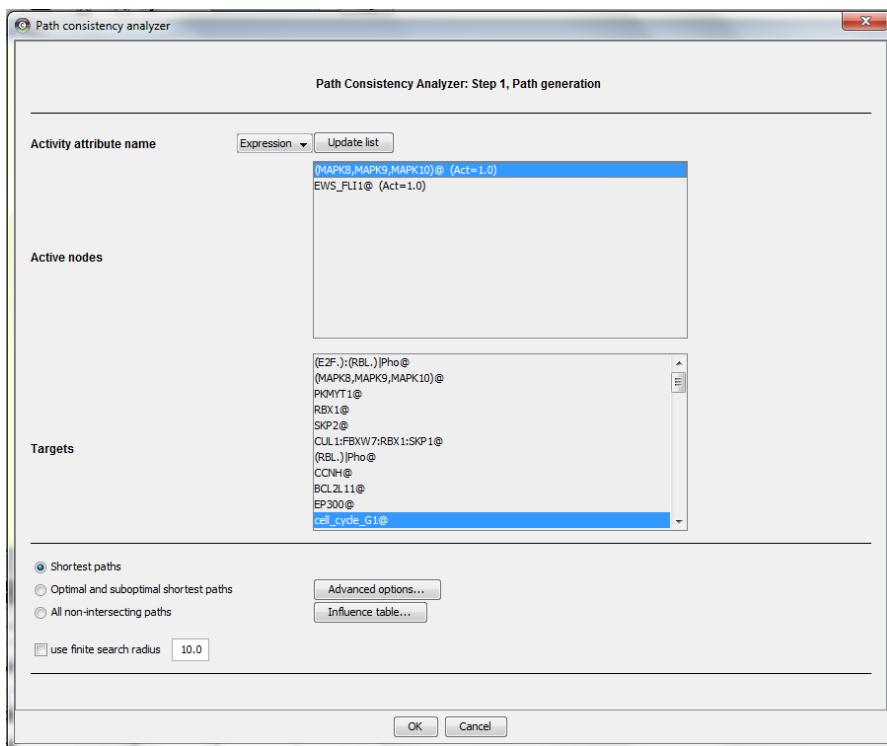


Figure 25: Path consistency analysis: dialog of step 1, select annotated nodes, targets and the algorithm for searching the paths.

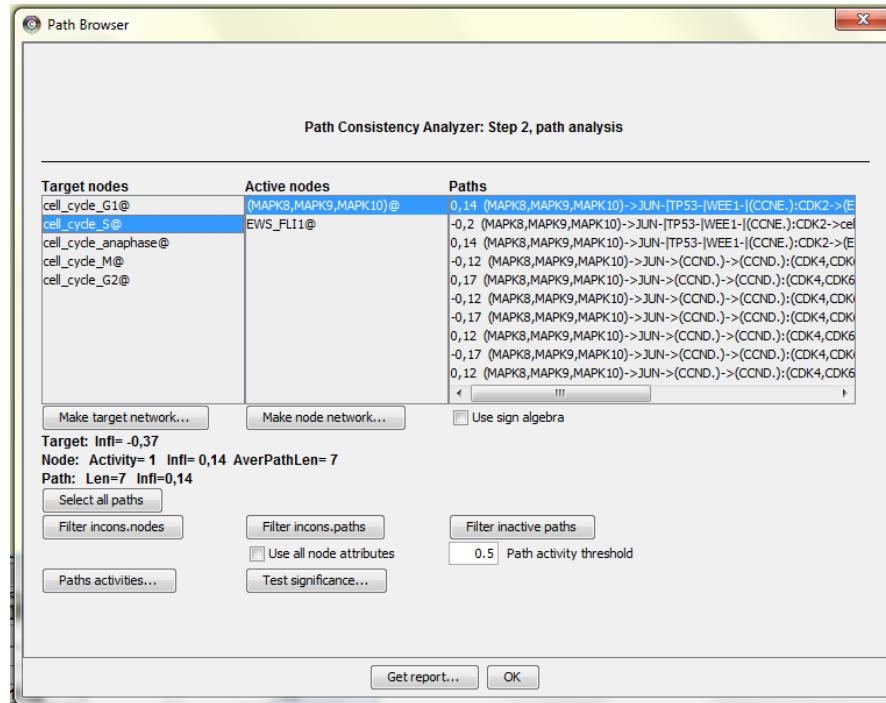


Figure 26: Path influence quantification analysis: dialog of step 2, showing all the results and available options.

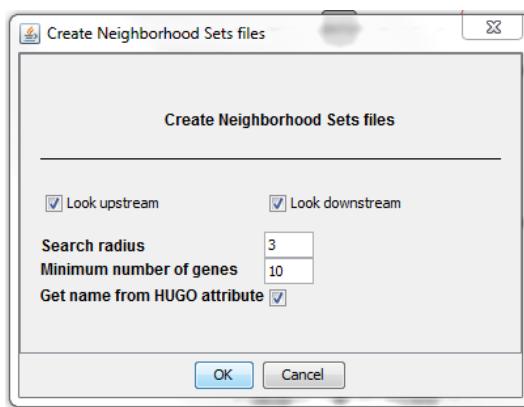


Figure 27: Dialog for setting the different options of creating a neighborhood sets file.

## 4 BiNoM Fade Signal Propagation Model

This menu provides functions of computing and visualizing influence according to a simple model describe in [??](#)). Some of these function needs parameters which are memorized

### 4.1 Preselect Sources and Targets

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Preselect Sources and Targets**

Preselect in dialogs nodes as sources or as targets. So, these nodes are selected, when opening list boxes, The node attribute PRESELECTED is set to 1 if source, to 2 if target. Not preselected node attribute is set to 0. It can be directly imported as node attribute by Cytoscape.

### 4.2 Select Sub-network from Sources to Targets

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Select Sub-Network From Sources to Targets**

This function selects nodes and edges in the network between a list of sources and a list of targets. Loops are included in the selection. For all nodes as sources or targets: select the first node and type shift+control+end.

### 4.3 Select Mono or Multi Path Model

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Select Mono or Multi Path Model**

This function selects the type of model: multi path or mono path (see [??](#)).

### 4.4 Display Signed Distances

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Display Signed Distances**

After selecting sources and targets, display the number of paths from sources to targets signed by weights according to the path model.

### 4.5 Update Influence Attribute

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Update Weigth Influence Attribute**

A weight of influence as edge attribute must be affected to every edge. This dialog updates the weight attribute by

- selecting the attribute containing the edge influence,
- affecting its values to activation (weight=+1) and inhibition (weight=-1).

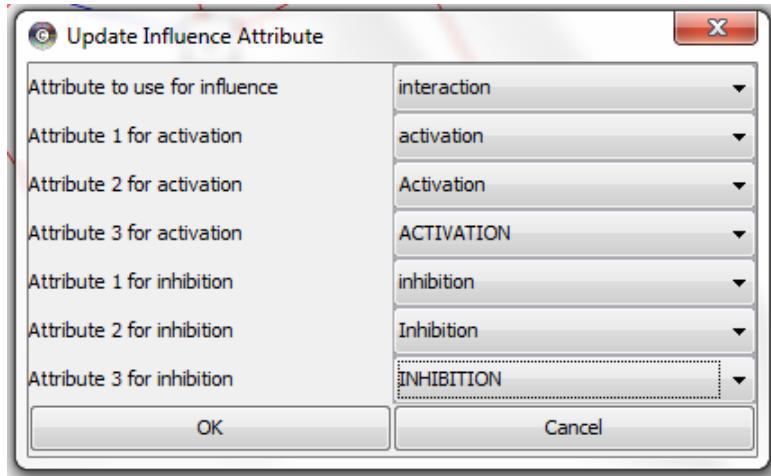


Figure 28: Dialog updating weight attribute from an other attribute

3 possible values for the attribute. Generally, the attribute is "interaction" and the values "activation" or "inhibition". Be careful of lower-case and upper-case (see ??).

#### 4.6 Input Reach Parameter

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Input Reach Parameter**

Input the number of paths beyond which the influence is insignificant, less than 5%. It is not necessary integer, it is double number.

#### 4.7 Display Network and Parameter Features

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Display Network and Parameter Features**

Display in a text box: network, size of network, reach parameter, min, max, mean and standard deviation influence, computed by excluding not connected nodes. It a recapitulation of parameters and their effect on influence matrix.

#### 4.8 List Opened Edges MultiPath Only

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒List Opened Edges MultiPath Only**

Only in multi path model. List edges which are opened to avoid loop and allow multi path computing.

## 4.9 Display Influence Array As Text

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Display Influence Array As Text**

The influence matrix is displayed in a text box which can be copied in the clipboard and paste in a spreadsheet, sources in columns, targets in rows, names in alphabetical order. Parameters and option are in window title.

Text window with 2 options:

- for visualizing, "nc" = not connected, only 3 digits after point for numbers,
- for computing, all values are numeric with all possible digits,

Same dialog as "Select Sub-network from Sources to Targets". Preselected sources and targets can be used. For all nodes as sources or targets: select the first node and type shift+control+end.

## 4.10 Display Influence Array as Paved Window

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Display Influence Array as Paved Window**

Same computing as "Display Influence Array As Text". Results are in:

- paved window with 2 options:
  - activated in red, inhibited in green, light to dark according to the value, not connected in black,
  - activated in red, inhibited in blue, light to dark according to the value, not connected in white (see ??),
- text window where are displayed details of selected area in paved window.

## 4.11 Influence by Active Nodes as Attribute

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Influence by Active Nodes as Attribute**

Activity levels of nodes are input in an attribute "ACTIV\_IN". The result of the multiplication of influence matrix by activity level input is "ACTIV\_OUT" attribute.

## 4.12 Display Influence Reach Area in Array

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Display Influence Reach Area in Array**

Computing as "Display Influence Array for Computing" with all weights=1. Useful to appreciate the absolute level of influence by a specie to other species.

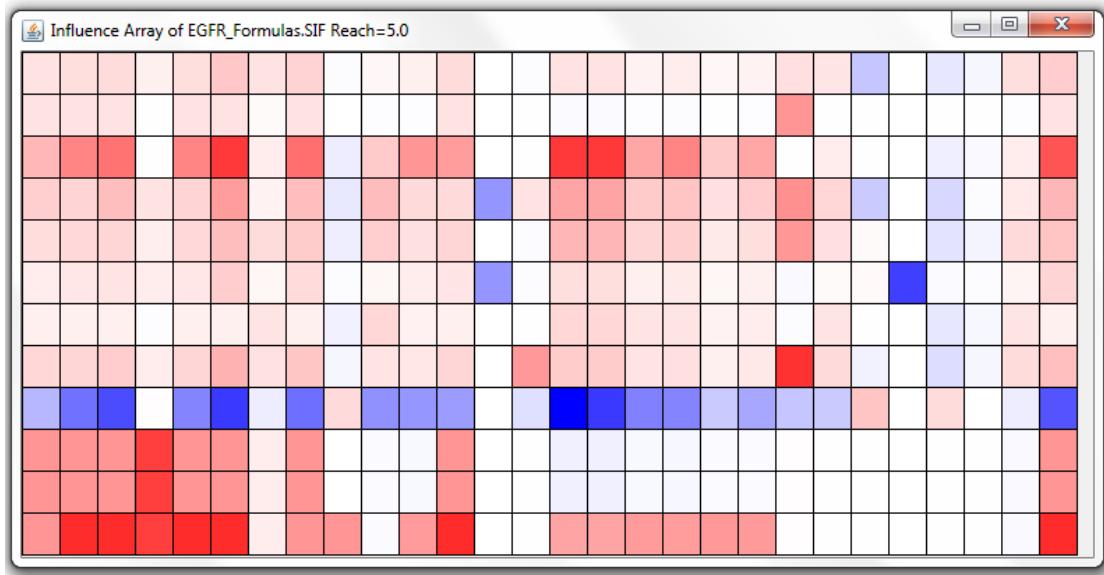


Figure 29: Window paved by the level of influence between species

#### 4.13 Influence Reach Area as Attribute

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Influence Reach Area as Attribut**

Computing from selected nodes with all weights=1. Absolute influence levels are put in attribute "INFLUENCE\_AREA\_N" where N keeps every successive results. Start nodes and options must be noted manually. Useful to visualize the influence of a group of species.

#### 4.14 Input Score Threshold

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Input Score Threshold**

Input the threshold used to compute the score.

#### 4.15 Compute Score of Data Sets

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Compute Score of Data Sets**

Data must be input as node attributes:

- Activity levels of source nodes as input, attribute "INPUT\_SETset\_identifier" (2 underscores), set\_identifier may be a number or a word,
- Expected activity levels of target nodes as output aim, attribute "OUTPUT\_AIMset\_identifier", set\_identifier matches input with output.

Set	Input Set Size	Output Aim Size	Sign Score	Active Ok	Inhibit Ok	Kappa
1	13	46	32	19	13	0.385496183
2	13	20	12	9	3	0.157894737
		66	44	28	16	0.320860617

Figure 30: Score of data sets formatted in a spreadsheet

The result is a tabulated text including size of data sets, number of matching output (with threshold) and Cohen's kappa coefficient (see ??). If level <-threshold, node is taken as inhibited. If level >+threshold, node is taken as activated.

#### 4.16 Test Score by Reversing Sign Weight

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Test Score by Reversing Sign Weights**

Test if reversing sign weight of every edge improves kappa, display list of edges sorted by decreasing kappa.

#### 4.17 Test Score by Canceling Weight

**Plugins⇒BiNoM⇒BiNoM Analysis⇒Fade Signal Propagation Model⇒Test Score by Cancelling Weight**

Test if cancelling sign weight of every edge improves kappa, display list of edges sorted by decreasing kappa. No change in structure network.

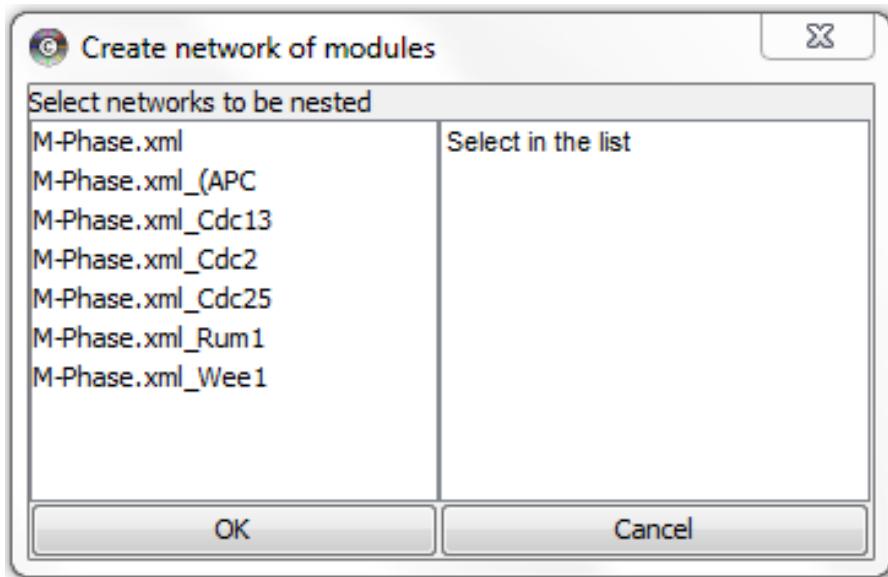


Figure 31: Dialog for selecting the networks that will be included in a module.

## 5 BiNoM Module Manager

The Module manager is useful for creating a modular view of large networks, i.e. a more compact network, without loosing any details of the initial global network. To achieve this goal, we are using the “nested networks“ feature of Cytoscape, introduced with the version 2.7.

### 5.1 Create Network of Modules

Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Create Network of Modules

This function creates a new network from a list of subnetworks (subnetworks are selected in the network list located on the left panel of the Cytoscape application, see figure ??). In the new network that is created, each node represents a module, pointing in fact to a subnetwork.



Module names, node names and network names must be different.

In order to visualize a subnetwork contained in a module, select the node corresponding to the module, then right-click with the mouse and choose ”Nested Network⇒Go to Nested Network“ from the contextual menu.

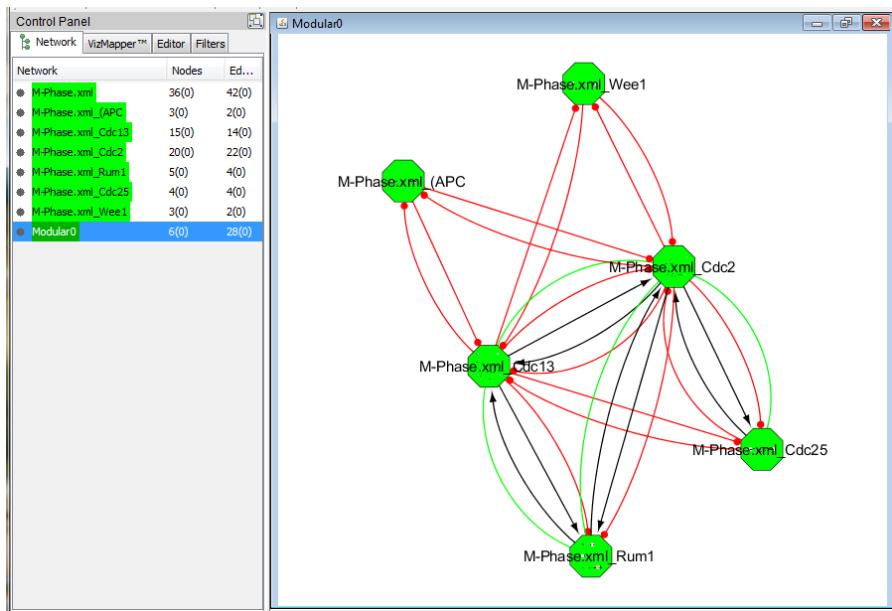


Figure 32: To generate this figure, the M-Phase network is divided into subnetworks by using the "get material component" function. The modular view is created by selecting different subnetworks and applying the function "create a network of modules". Then, to create the links between the modules, the function "Create connections between modules" links modules according to the reference global network. The function "Find common nodes in modules" creates intersection edges.

## 5.2 Create Connections between Modules

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Create Connections between Modules**

This function creates edges linking modules from all the edges of the selected network. The links are simplified, no distinction is made between left and right (molecule flow), there is no duplication if two interactions are the same.

A warning message is displayed if there are duplicated or absent nodes.

## 5.3 Pack Network In Modules

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Create Modules from Networks**

This function creates modules in a copy of the active network from a list of sub-networks (sub-networks are selected in the network list). All edges are kept and duplicated if necessary by making an edge ID from those in the initial network

The attribute BIOPAX\_NODE\_TYPE is set to pathway (see visual style BiNoM BioPAX).



All nodes of sub-networks must be found in the active network.

© Agglomerate the nearest nodes in modules X

Create 5 modules with parameters: 1/12

Name	Size	LastDistance	LinkNumber
Module_1/12_01	6	1	1
Module_1/12_02	3	1	1
Module_1/12_03	8	1	1
Module_1/12_04	9	1	1
Module_1/12_05	10	1	1

Copy whole to Clipboard Yes No

Figure 33: This window displays modules, number of node, last distance and number of links of agglomerating process. Yes lauches the process of agglomerating.

#### 5.4 Agglomerate the Nearest Nodes in Modules

Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Agglomerate the Nearest Nodes in Modules

This function creates modules and a modular view by agglomerating the nearest nodes in the active network (see the algorithm description in section ??).

There are two input parameters in order to limit the computational time and the number of modules:

- Maximal distance between nodes or modules in number of edges,
- Maximal number of nodes in modules.

Confirm creation if agree with displayed result (see dialog ??).

Sub-networks are created and the corresponding network of modules. In it, nodes are links by creating connections between modules (see figure ??).

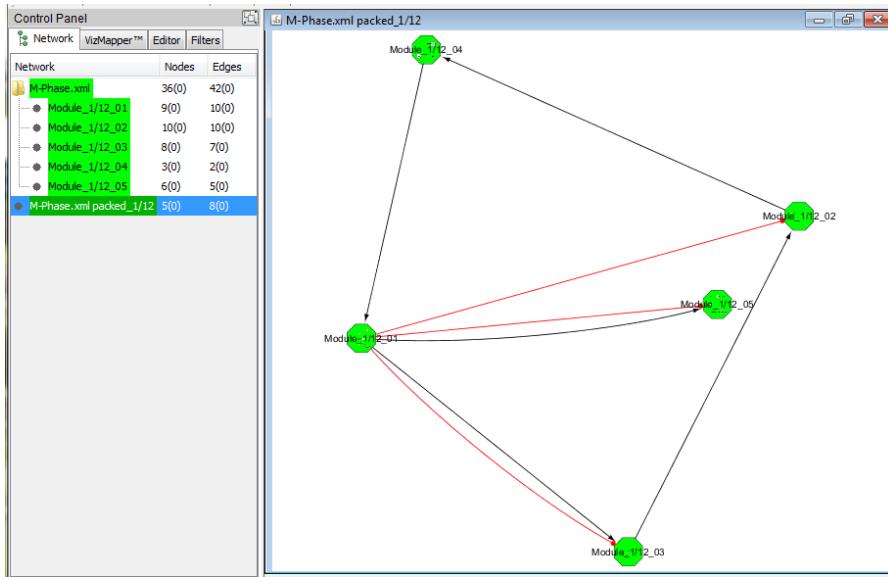


Figure 34: M-Phase is got by creating network from modules, modules created by agglomerating the nearest nodes (maximal distance=1, maximal size=12 nodes).

## 5.5 List Nodes of Modules and Network

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒List Nodes of Modules and Network**

This function list the nodes of the global network and nodes included in modules. The results in the text box can be simply copied in a spreadsheet through the clipboard.

## 5.6 Find Common Nodes in Modules

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Find Common Nodes in Modules**

Display in a text box the matrix of nodes (modules in columns, nodes in rows, size of modules in last row, frequency in modules in last column). The result is more easily usable after copying it in a spreadsheet. (see ??).

Create intersection edges with number of common nodes as attribute (COMMON\_NODES), green edges in figure ??.

Create a node attribute containing the node numbers of modules (NODE\_NUMBER).

The Module Visual Style can be adapted to the wished visual aspect by hand in the Cytoscape VizMapper, for example:

- To visualize NODE\_NUMBER: double click Node Size, select NODE\_NUMBER,

	M-Phase.xml _Wee1	M-Phase.xml _Rum1	M-Phase.xml ..._Cdc25	M-Phase.xml _Cdc2	M-Phase.xml _Cdc13	M-Phase.xml _(APC)	Frequency
(APC:S:p1) active	0	0	...	0	0	1	1
Cdc13:Cdc2 Thr14_phos Tyr15_phos Thr167_phos	0	0	...	0	1	1	2
Cdc13:Cdc2 Tyr15_phos	0	0	...	0	1	1	2
Wee1	1	0	...	0	0	0	1
Cdc13:Cdc2	0	0	...	0	1	1	2
Wee1:pho	1	0	...	0	0	0	1
Cdc13:Cdc2 Thr167_phos Rum1	0	1	...	0	1	1	3
...	...	...	...	...	...	...	...
Cdc25 pho active	0	0	...	1	1	0	2
re3	0	0	...	0	1	1	2
(APC:S:p1')  active' active	0	0	...	0	0	0	1
Cdc13	0	0	...	0	0	1	1
re15	0	0	...	1	1	0	2
re2	0	0	...	0	1	1	2
re20	0	1	...	0	0	0	1
Size	3	5	...	4	20	15	3

Figure 35: Matrix of nodes: modules in columns, nodes in rows, size of modules in last row, frequency in modules in last column.

continuous mapping, adjust width by graphical view.

- To visualize COMMON\_NODES double click Edge Line Width, select COMMON\_NODES, continuous mapping, adjust width by graphical view.

## 5.7 Assign Module Names to Node Attribute

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Assign Module Names to Node Attribute**

Function to create a node attribute (named as the modular network), containing module names. This attribute may be used to visualize modules in the reference network.

## 5.8 List Components of Species in Network and Modules

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒List Components of Species in Network and Modules**

Function to list the different components of a given species (their names must respect BiNoM syntax). This can be useful to name modules.

## 5.9 Create Network from Union of Selected Modules

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Create Network from Union of Selected Modules**

Function to create a network from the union of selected modules and its corresponding module in the current network (named by module names separated by &).

## 5.10 Create Network from Intersection of 2 Selected Modules

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Create Network from Intersection of 2 Selected Modules**

Function to create a network from the intersection of 2 selected modules and its corresponding module (named by module names separated by |).

The user must confirm for deleting the common nodes in the selected modules.

### **5.11 Recreate Lost Connections inside Modules**

**Plugins⇒BiNoM 2.1⇒BiNoM Module Manager⇒Recreate Lost Connections Inside Modules**

Function to recreate the connections inside modules which may have been lost by modularizing operations.

### **5.12 Destroy Networks Unused as Module**

**Plugins⇒BiNoM 2.1⇒BiNoM module manager⇒Destroy Networks Unused as Module**

Select networks to be deleted among a list of networks which are not used as modules in the current network (simplify cleaning session).

## 6 BiNoM BioPAX3 Utils

### 6.1 BioPAX 3 Property Editor

Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Utils⇒ BioPAX 3 Property Editor

All the information available in a BioPAX file can be easily retrieved using the BioPAX Property Editor function. A component on the diagram must be selected first (CDC2 in Figure ??) and a window appears with all available information concerning the molecule.



Figure 36: BioPAX Property Editor: example of the properties concerning CDC2 component in M-Phase model

In the menu of the Property Editor, several options are offered:

- Display valid attributes / Display all attributes hides all the empty fields (for example, in Figure ??: Availability or Evidence have <empty object list>and would be hidden) / shows all the available fields, even the empty ones.
- << and >> correspond to back or forward buttons and follow the historical exploration of the Property Editor (similar to Back and Forward buttons of a network browser).
- Close current tab or Close all tabs closes the current page of the property editor or all the open pages.
- Display / Edit shows a simple display of the page editor where no change can be made (Figure ??) / allows changes in the fields by adding, removing or updating information (Figure ??). For the latter, click first on the Edit tab on the upper menu, then on update situated near the field to modify. In Figure ??, as an example, some comments were added manually: CDC2 is a kinase that binds to CDC13 to form a dimer. In the Apoptosis example (Figure ??), extensive information is already available concerning the pathway, references, etc.

For more details on BioPAX description standard, visit the webpage: <http://www.biopax.org/>

## 6.2 BioPAX 3 Class Tree

Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Utils⇒BioPAX 3 Class Tree

All the statistics concerning the pathway are listed: the number of reactions, associations or catalyses, the number of proteins or complexes, etc (figure ??). More information can be accessed by selecting a specific object which, when clicked on, leads to the BioPAX 3 Property Editor window (see section ??).

To complete the network, the user can easily add new information or a new protein, protein complex, type of interaction, etc., by clicking on the New Instance tab.

## 6.3 Use Simplified URI Names

Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Utils⇒Use Simplified URI Names

In the BioPAX Class Tree, protein names can have either URI names (Uniform Resource Identifier used to give a unique identification to proteins) or BiNoM Naming Service names. For example, for the apoptosis pathway, the protein BAD is referred to as

*UniProt\_Q92934\_Bcl2\_antagonist\_of\_cell\_death\_BAD\_\_Bcl\_2\_binding\_component\_6\_\_Bcl\_XL\_Bcl\_2\_associated\_death\_promoter\_Bcl\_2\_like\_8\_protein*



Figure 37: BioPAX Property Editor: example of apoptosis pathway node properties

in the URI case and just BAD in the BiNoM Naming Service case. For the rules of how BiNoM generates names see section ??.

## 6.4 Synchronize networks with BioPAX 3

Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Utils⇒Synchronize networks with BioPAX 3

This command updates all the interfaces according to the changes made to individual BioPAX objects.

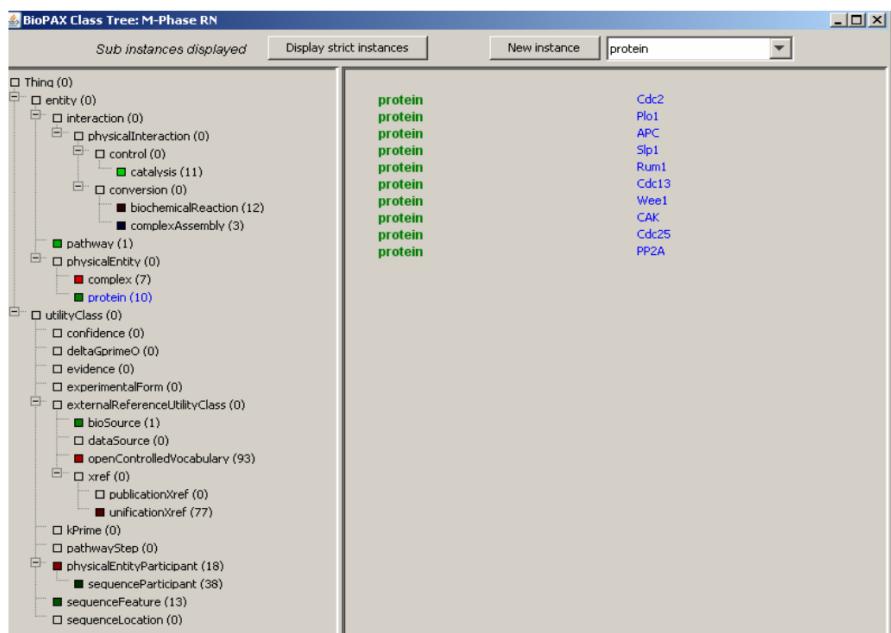


Figure 38: BioPAX Class tree. On the left frame, the model is described in terms of interactions, entities, etc. On the right frame, the proteins, selected in the left frame, are listed. The links are clickable and open a BioPAX Property Editor window.

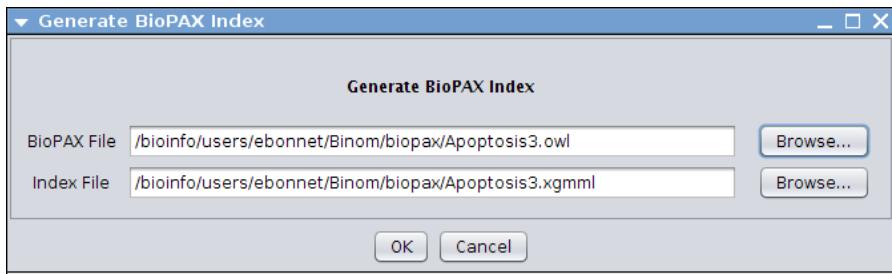


Figure 39: Dialog for generating a BioPAX Index.

## 7 BiNoM BioPAX3 Query

The purpose of the functions related to the query language is to work with huge BioPAX files and extract from the BioPAX documents only the information that is of interest. For this part, we will use the apoptosis example initially extracted from Reactome database: Apoptosis3.owl (extracted from the Reactome database, available from our website). This set of functions can be used with big pathway databases already exported to BioPAX, such as Reactome, BioCyc, NetPath (see <http://www.biopax.org> for the complete list).

### 7.1 Generate Index

**Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Query⇒Generate Index**

Using this function BiNoM maps the content of BioPAX file onto a labeled graph (referred to as index). It creates an \*.xgmml file from an \*.owl one (figure ??). For the definition of BioPAX index, see section ??.

### 7.2 Load Index

**Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Query⇒Load Index**

Once the xgmml is created, it can be loaded into memory. The index is a global object, i.e. only one index can be used at a time. The function “Load Index” loads the index file from xgmml format (figure ??).

Together with the index, you can also upload a tab-delimited accession number file which corresponds to a list of synonyms for the genes/proteins ids used in a network (see an example of the content of some accession number file at figure ??). An entity in the index can be identified by its id, by any XREF attribute (see section ??), by node name, or by any synonym from the accession table (if it is provided).

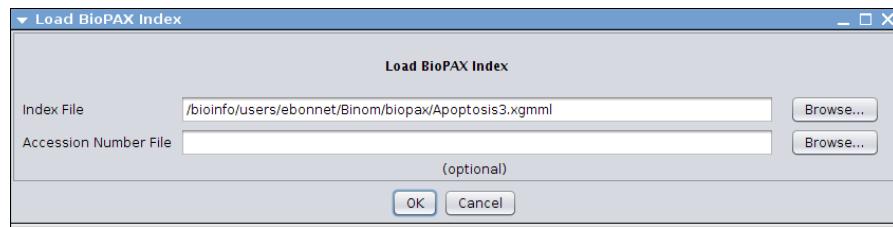


Figure 40: Load Index dialog.

cAMP-dependent protein kinase	ID001
cAPK	ID001
InterPro:IPR000719	ID001
PKA catalytic subunit	ID001
PKAc	ID001
PKAc-beta	ID001
Protein kinase	ID001
protein kinase A	ID001
protein kinase A catalytic subunit	ID001
protein kinase A catalytic subunit	ID001
Protein kinase C-terminal domain	ID001
Serine/threonine protein kinase	ID001
Serine/threonine protein kinase, active site	ID001
Tyrosine protein kinase	ID001
ADP-ribosylation factor	ID002
ADP-ribosylation factor family	ID002
ARF	ID002
Rab	ID002
Ran family	ID002
Ras family	ID002
Rho family	ID002
Rho/Rac family	ID002
small G-proteins	ID002
small GTPases	ID002
small guanyl-nucleotide binding proteins	ID002
activator of G-protein signaling	ID003
AGS	ID003
InterPro:IPR001806	ID003
InterPro:IPR001806	ID003
InterPro:IPR003577	ID003
InterPro:IPR005225	ID003
InterPro:IPR005225	ID003
p21ras	ID003
Rad	ID003
Ral	ID003
Rap	ID003
Ras	ID003
Ras family	ID003
Ras GTPase superfamily	ID003
Ras GTPase superfamily	ID003
Ras homolog enriched in brain	ID003
Ras small GTPase, Ras type	ID003

Figure 41: Example of accession Number file. First column is a synonym (which can have structure  $<database> : <standard_id>$ ), the second column is the id used inside the BioPAX file.

### 7.3 Display Index Info

Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Query⇒Display Index Info

This command opens a window indicating the name of the graph, the name of the file, the accession number file, when available, the number of records, and the various statistics of the index: number of publications, proteins, physicalEntities, complexes, biochemical reactions, pathways, pathwaySteps, catalyses, and modulations (necessary proteins for catalyses). See figure ??.

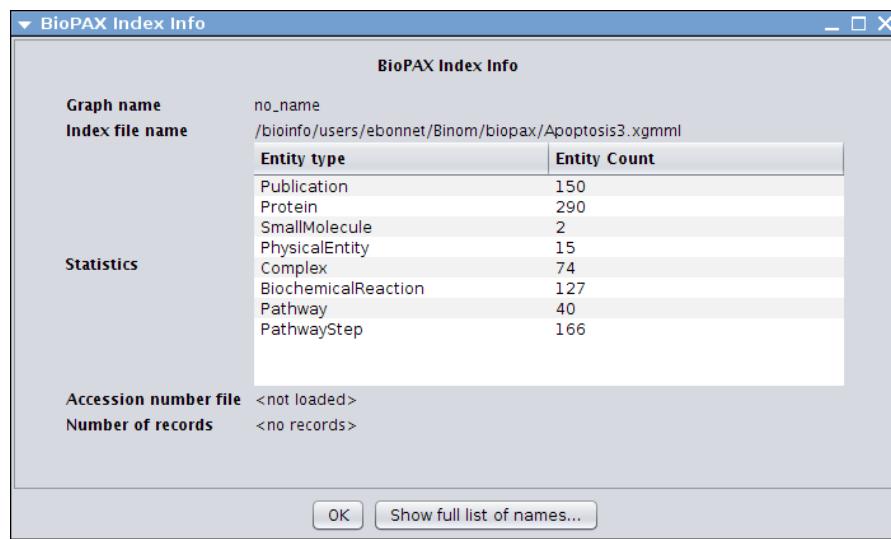


Figure 42: Display Index info.

### 7.4 Select Entities

Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Query⇒Select Entities

The BioPAX document is often too big to find the protein or gene that needs to be studied. To access it easily and rapidly, it is possible to find the component directly with this command and build a specific network around that molecule.

For example, in the Apoptosis3 network, we choose to find the SMAC protein and extend the network around it. A dialog window pops up and offers the possibility to find a protein or a gene by its name or id or XREF attribute or synonym, from the current network when a network is already opened, or from the list of identities associated with the BioPAX index (figure ??).

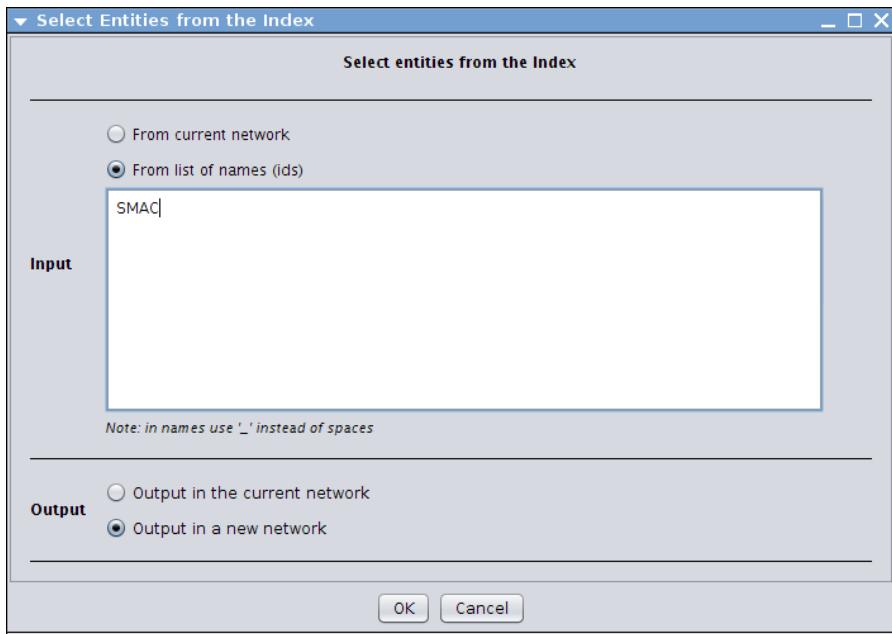


Figure 43: Select entities from the index.

For our example, we choose the second option. Note that it is possible to type several versions of the protein name (separated by space, comma, or semi-colon symbols). A new network is created, having only one node (SMAC). It is also possible to select more than one entity, in this case, the components all appear in the same window.

It is also an option to view several genes or proteins in the same network by checking output in the current network.

Note that it is advised to use the BiNoM BioPAX visual style to view the resulting network.

## 7.5 Standard Query

**Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Query⇒Standard Query**

This command proceeds through a series of actions that will extend the initial network.

Lets start with diverse queries from the network created from the Apoptosis file with the initial SMAC entity. A dialog window opens as figure ??.

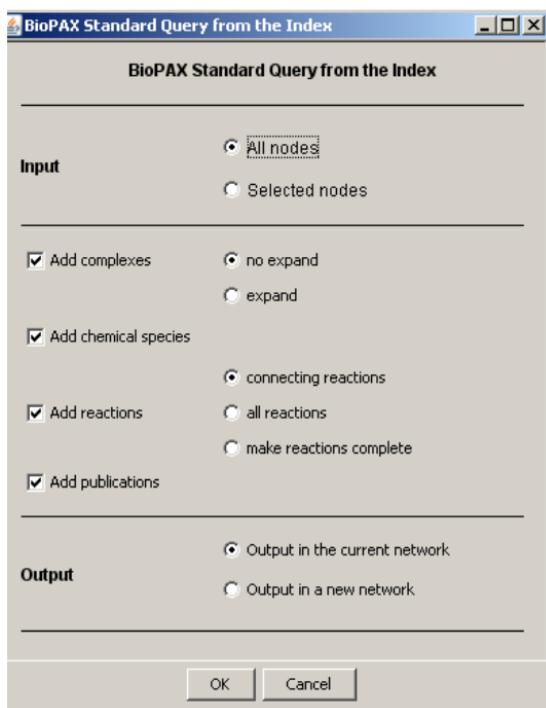


Figure 44: BioPAX Standard Query: dialog window.

All the options proposed in the dialog window (figure ??) are listed here:

### 7.5.1 Input

In the network, you can submit queries that concern all the nodes (option All Nodes) in the network or only the selected nodes (option Selected nodes).

### 7.5.2 Adding nodes

- Add complexes
  - The option no expand adds only the homodimers of the molecule. If several proteins were queried, then all hetero-dimers in which all the proteins participate would be selected.
  - The option expand adds all the complexes in which SMAC is involved (figure ??b). The green arrow with a diamond ending represents the inclusion of one protein in a complex form.
- Add chemical species
 

This function adds, for each species, the cellular location and its specified modifications.

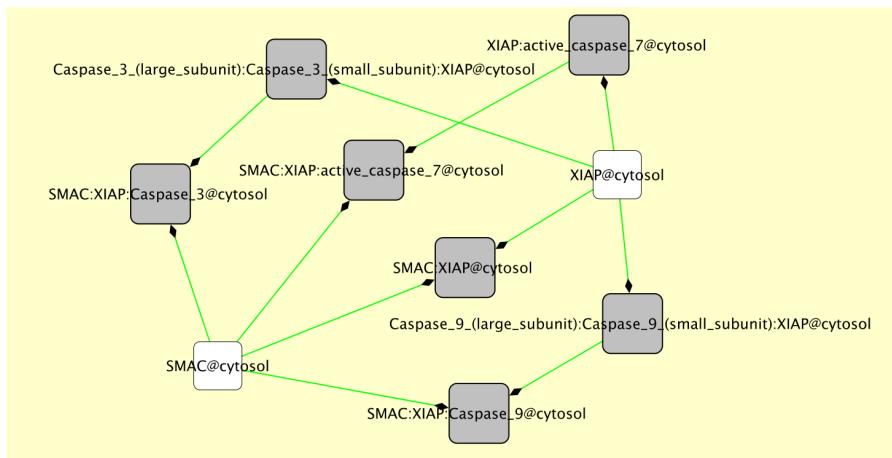


Figure 45: BioPAX Standard Query: Complexes involving SMAC are added to the network.

- Add reactions
  - The option "connecting reactions" connects all present species that have common reactions (Figure ??).
  - The option "all reactions" includes all the reactions involving the chemical species.
  - The option "make reactions complete" adds all the sources and targets of the reactions listed in the BioPAX index, including the pathway nodes and publications links.
- Add publications
 

When available, this function adds all the references associated with a reaction (see figure ??).

### 7.5.3 Output

The result of the queries can be seen either in the current network or in a new network.

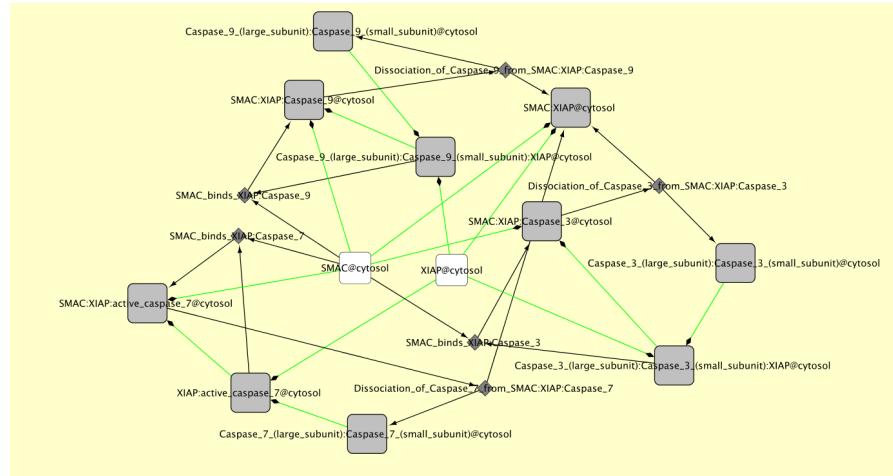


Figure 46: Adding reactions: SMAC centered network expanded with the connecting reactions.

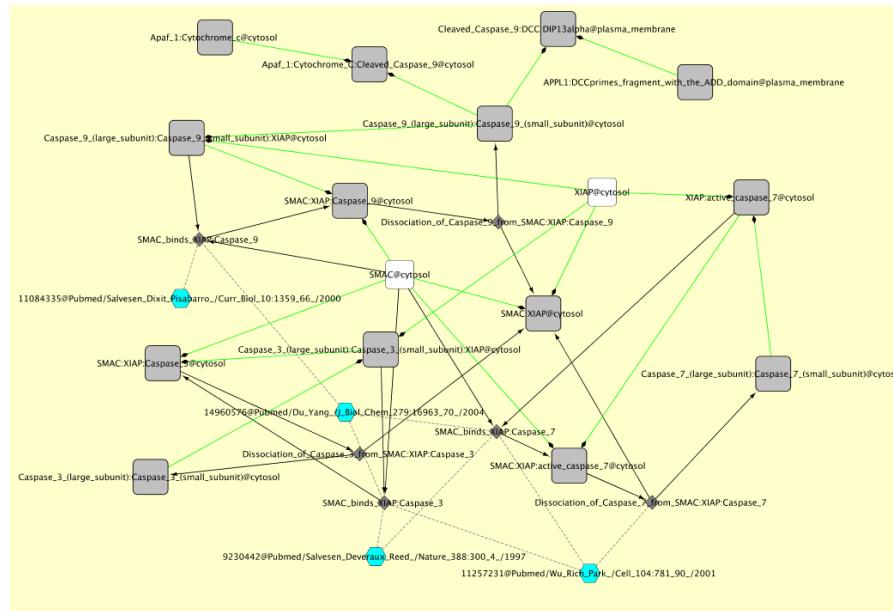


Figure 47: Adding publications.

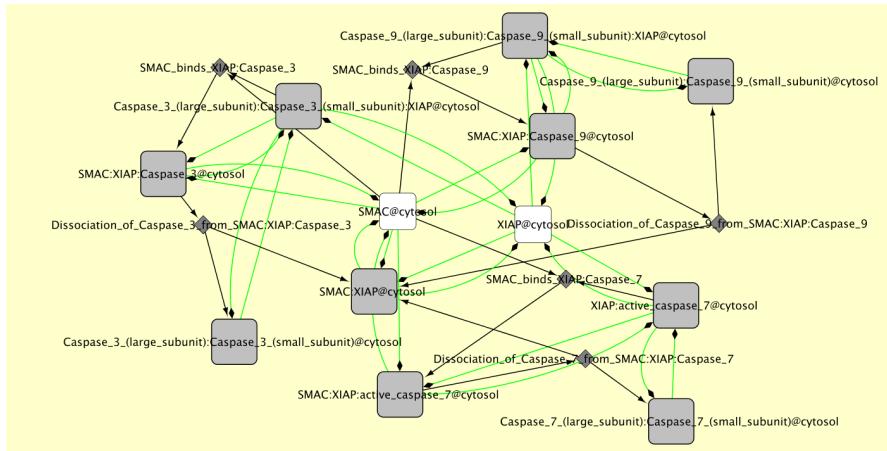


Figure 48: Index path Analysis with SMAC and XIAP.

## 7.6 Index Path Analysis

**Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Query⇒Index Path Analysis**

This command finds the directed or non-directed, shortest, optimal or suboptimal, non intersecting paths with a pre-defined number of intermediaries in an index file. Note that the species need to be selected on a graph before this query.

This part of the query engine uses the same algorithms and options as Path analysis dialog (see section ??), however, with the network (index) kept completely in memory, without explicit visualization. Moreover, the network is slightly modified before this type of query: in particular, all non-directed edges (of CONTAINS, SPECIESOF and some other types) are represented as bi-directional, some nodes (publications and, optionally, smallMolecules) are removed.

For example, the following steps

1. Select Entities: specify SMAC and XIAP proteins.
2. Select the two nodes.
3. Index Path analysis: Find all non-intersecting paths.
4. BiNoM Analysis: Extract Reaction Network.

will produce the following network connecting SMAC and XIAP proteins (Figure ??).

## **7.7 View Query Log**

**Plugins⇒BiNoM 2.1⇒BiNoM BioPAX 3 Query⇒View Query Log**

In this window are recapitulated all the queries done during the session.

## 8 BiNoM Utilities

There are various functions that facilitate the manipulation of the networks, the copying and pasting of its subparts, the selection of some portions of it, etc. Those of them are missing in Cytoscape.

### 8.1 Select Edges between Selected Nodes

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Select Edges between Selected Nodes** or f8

When some components are selected by their names (Select⇒Nodes⇒By Name) or simply with the mouse, the edges between the nodes are not selected. This function allows to remedy this problem.

This might be especially useful when the selection is copied and pasted in another network, although it is possible to paste the nodes with the edges connecting them, without selecting them, by choosing File⇒New⇒Network⇒From selected nodes, all edges when pasting them.

### 8.2 Select upstream neighbours

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Select upstream neighbours** or ctrl+8  
Select upstream neighbours of selected nodes. Whole upstream neighbours can be selected by repeating the command until no more node is selected.

### 8.3 Select downstream neighbours

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Select downstream neighbours** or ctrl+9  
Select downstream neighbours of selected nodes. Ditto downstream.

### 8.4 Double Network Differences

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Double Network Differences**

A network is composed of nodes and edges. With this command, two networks A and B can be compared and the difference observed between the two is created in two new graphs: A-B for the differences of A compared to B ( $A-A\cap B$ ), and B-A for the differences of B compared to A ( $B-A\cap B$ ). Note that in the output networks, the layout of the first graph is conserved.

### 8.5 Update Networks

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Update Networks**

When a network is modified, it is possible to update all the networks that are related to it, either because they are modules, sub-networks or older versions of it. That way, any changes, additions or deletions in a network can be propagated to the sub-parts that

are derived from the initial version of that network. The user specifies which network is added, which one is deleted (when necessary) and which networks need to be updated in the proposed list.

Whatever is added or deleted in each sub-network is presented to the user in a separate window before any action is made. The user can agree or disagree by checking or unchecking the box in front of the network names. The previous version of the updated diagrams will not be deleted but saved under *network\_name.xml.old*.

## 8.6 Update connections from other network

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Update connections from other network**

The dialog (figure ??) propose to select:

- From network, the reference network where are the connections.
- Networks to update: networks in which connections are copied.

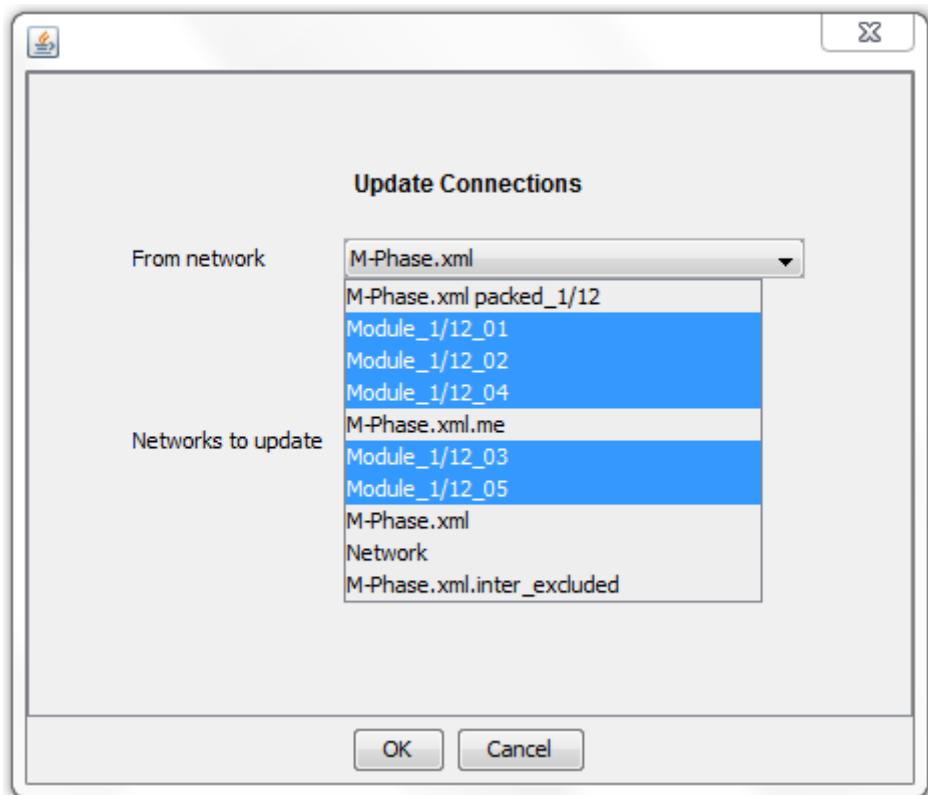


Figure 49: Update connections from other networks Dialog

Some connections may be duplicated. So a Cytoscape command (version 8.0) may be used to delete them:

**Plugins⇒Network Modifications⇒Remove Duplicated Edges**

## 8.7 Merge Networks and Filter by Frequency

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Merge Networks and Filter by Frequency**

Create a network by merging the selected networks where the percentage of common nodes is greater than intersection threshold (see dialog figure ??).

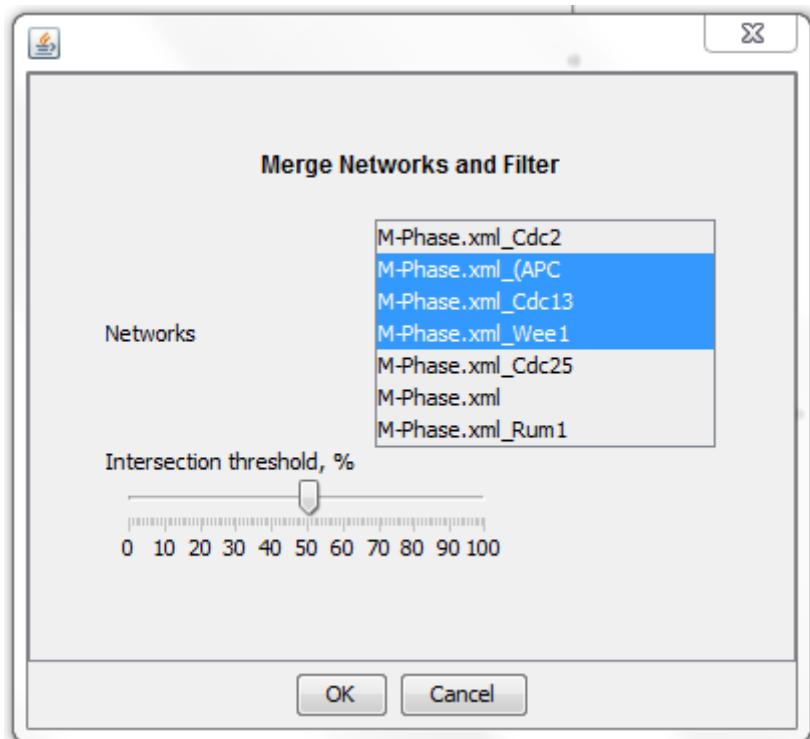


Figure 50: Merge Networks and Filter by Frequency Dialog

## 8.8 Clipboard

The description of the commands, which work only inside a Cytoscape session, is self-explanatory.

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Clipboard⇒Copy selected nodes and edges to clipboard**

**Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Clipboard⇒Add selected nodes and**

**edges to clipboard**

Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Clipboard⇒Paste nodes and edges from clipboard

Plugins⇒BiNoM 2.1⇒BiNoM Utilities⇒Clipboard⇒Show clipboard contents

## 9 Appendices

### 9.1 Attributed graph model

BiNoM manipulates the information contained in the standard systems biology files by mapping it onto a labeled graph, called index. The index does not try to map the totality of all details; it rather serves as a connection map for the objects contained in other ontologies such as BioPAX. In other words, the index contains the minimum information needed to graphically represent objects and connections between them. Index elements (nodes and edges) are annotated by identifiers sufficient to find these objects in the original files and extract and edit the information related to them.

This approach has several advantages, in particular, with respect to synchronization issues. BiNoM index is a light-weight construction which can be easily regenerated, does not duplicate the information in existing files and serves only to facilitate the visualization and to access existing systems biology files.

Currently, BiNoM index is mostly developed to map BioPAX ontology files and CellDesigner object schema. In future versions, other mappings will be available, for instance, a mapping to SBML files annotated with Systems Biology Ontology (<http://www.ebi.ac.uk/sbo/>).

The table ?? lists all attributes used by the index.

### 9.2 BiNoM CellDesigner and BiNoM BioPAX visual mappers

BiNoM has two built-in visual mappers supporting the visualization of the whole index or of its parts. The legend for deciphering the different types of visualization is provided in figure ??.

### 9.3 BiNoM Naming Service

When importing pathway information, BiNoM tries to generate meaningful, unique and short names for index entities. This function of the plugin is performed via BiNoM Naming Service. For proteins and other entities, the shortest available synonym is used. For genes, a g symbol is added at the beginning of the name, and for RNAs, a r symbol is added in order to avoid mixing genes and mRNAs with their products. If this leads to an ambiguity, it is resolved by adding a suffix specifying a unique id of the entity.

A chemical species in BiNoM is defined as a physical entity (such as protein) with some cellular localization and some (post-translational) modification (possibly none). The general template of the species label is the following:

Entity1\_name|Modification1|Modification2|:Entity2\_name|Modifications...[.active|.hmN]@compartment

Here, the colon symbol : delimits the different components of a complex if the species has several components. Optional suffixes active or hm describe active state of the chemical species or N-homodimer state, respectively.

Node attributes		Edge attributes	
Attribute name	Meaning and possible values	Attribute name	Meaning and possible values
<b>BioPAX attributes</b>			
BIOPAX_NODE_TYPE	Type of entity represented by the node: protein, dna, rna, smallMolecule, transport, biochemicalReaction	BIOPAX_EDGE_TYPE	Type of connection between nodes: LEFT, RIGHT, CATALYSIS, NEXT, REFERENCE
BIOPAX_SPECIES	Species unique name	BIOPAX_EDGE_ID	Unique edge id
BIOPAX_REACTION	Reaction id	BIOPAX_URI	Full URIs of the objects associated with the edge
BIOPAX_NODE_SYNONYM	All entity synonyms		
BIOPAX_NODE_XREF	All entity accession numbers		
BIOPAX_URI	Full URIs of the objects associated with the node		
<b>CellDesigner attributes</b>			
CELLDESIGNER_NODE_TYPE	Type of entity represented by the node: PROTEIN, GENE, STATE_TRANSITION...	CELLDESIGNER_EDGE_TYPE	Type of connection between nodes: such as LEFT, RIGHT, CATALYSIS
CELLDESIGNER_SPECIES	For species nodes -- SBML species id		
CELLDESIGNER_REACTION	For reaction nodes -- SBML reaction id		
CELLDESIGNER_ALIAS	CellDesigner node alias		
<b>Common attributes</b>			
EFFECT	For reactions and influence edges -- effect of the influence. If something is known, such terms as 'activation', 'inhibition', 'catalysis' should be used		

Figure 51: All attributes of graph model used by the index

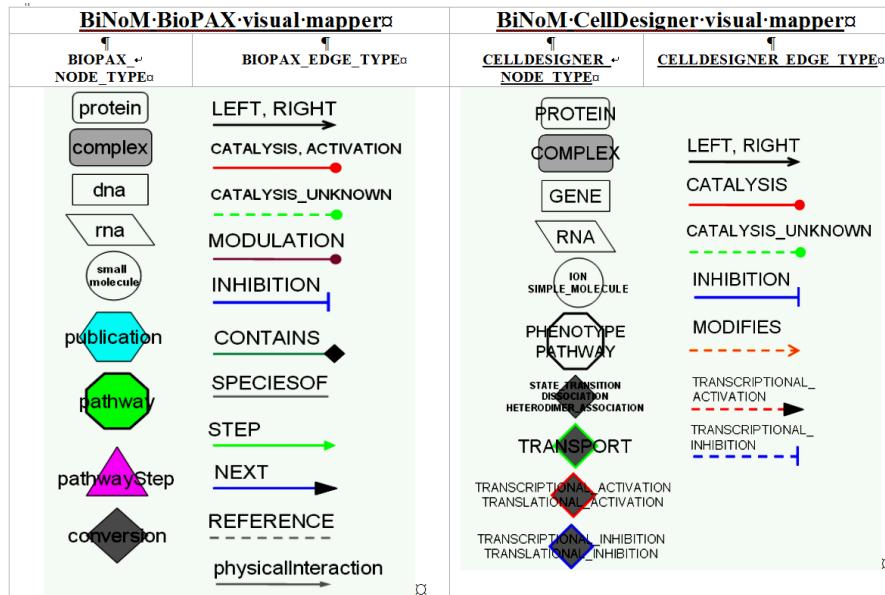


Figure 52: Types of visualization in BioPAX and CellDesigner

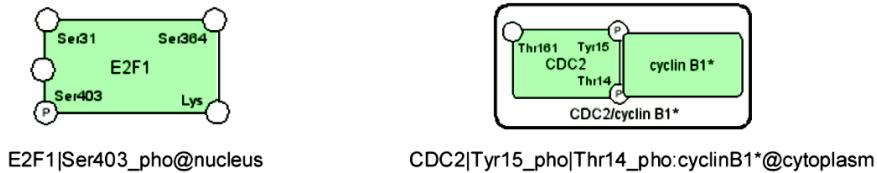


Figure 53: 2 examples of naming chemical species shown in Systems Biology Graphical Notation standard.

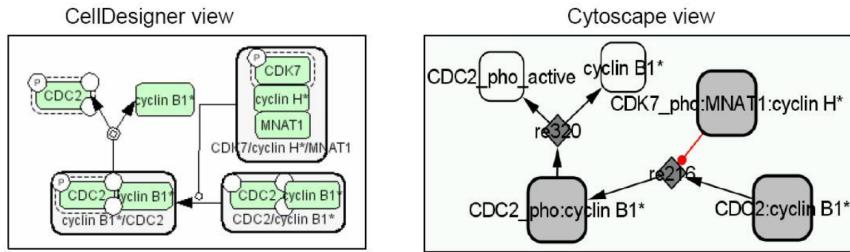


Figure 54: Conversion of a little network from CellDesigner Graphical Notation to BiNoM index representation

Several examples of naming chemical species are presented:

- Naming chemical species shown in Systems Biology Graphical Notation standard figure ??
- A conversion from CellDesigner figure ??.
- A conversion from BioPAX figure ??.

#### 9.4 Standard BioPAX interfaces

BiNoM index serves as a visual connector to the content of a network file. However, with all types of relations, the index is a highly connected graph and not very insightful when represented entirely. A subgraph of the index can be extracted according to a specific purpose and used to understand a specific aspect of the pathway information. We will call interface such a subgraph of the entire index.

When importing a BioPAX file, BiNoM proposes to generate three standard BioPAX interfaces referred to as

- Reaction Network.
- Pathway Structure.
- Protein-Protein Interaction.

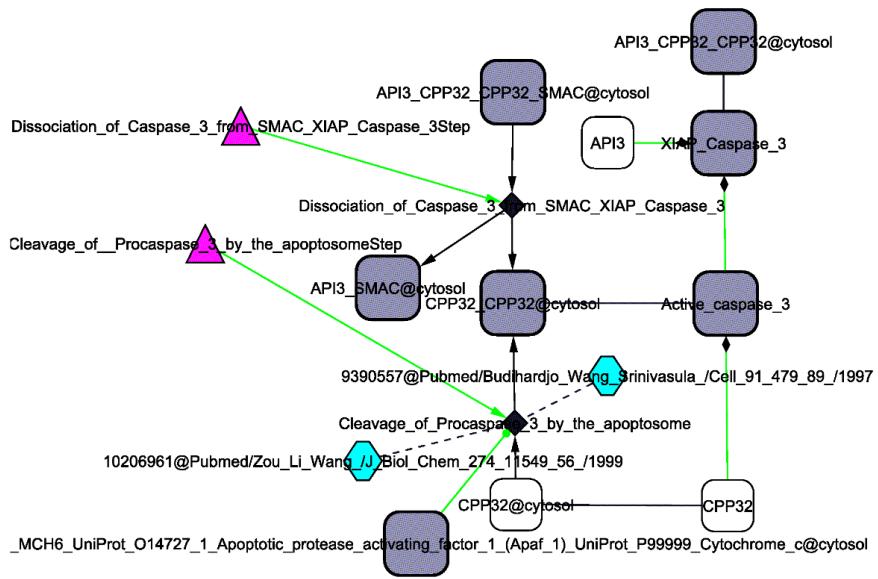


Figure 55: Small fragment of BioPAX index generated for Apoptosis pathway and extracted from Reactome database

#### 9.4.1 BioPAX interface as Reaction Network

The Reaction Network interface is a bipartite graph which contains nodes of only two types: species and reactions. Reactants are connected to reactions through edges of type LEFT, products are connected through edges of type RIGHT. Modifier species are connected through CATALYSIS, MODULATION and other edges. See figure ??.

Some BioPAX objects (catalysis, for example) are represented by edges with the corresponding BIOPAX\_URI attribute. A chemical species node can correspond to several grouped physicalEntityParticipants, thus, it can have several BIOPAX\_URI attributes. When calling BioPAX editor, all of them will be opened.

Standard Reaction Network interface can be exported to pure SBML format (level 2) and serve as a draft for further computational modeling.

#### 9.4.2 BioPAX interface as Pathway Structure

Pathway Structure interface contains only nodes of pathway, pathwayStep and interaction types. The types of the edges connecting them are CONTAINS, STEP and NEXT. See figure ??.

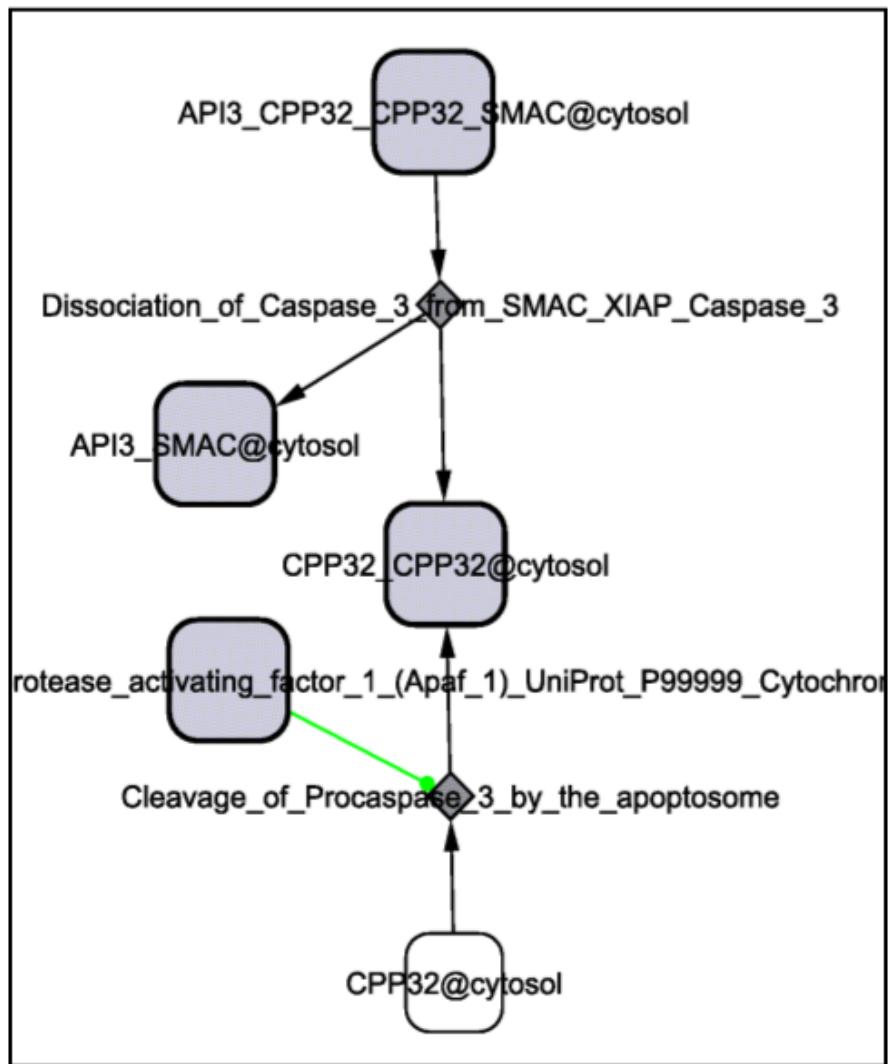


Figure 56: Fragment of Apoptosis from Reactome as Reaction Network.

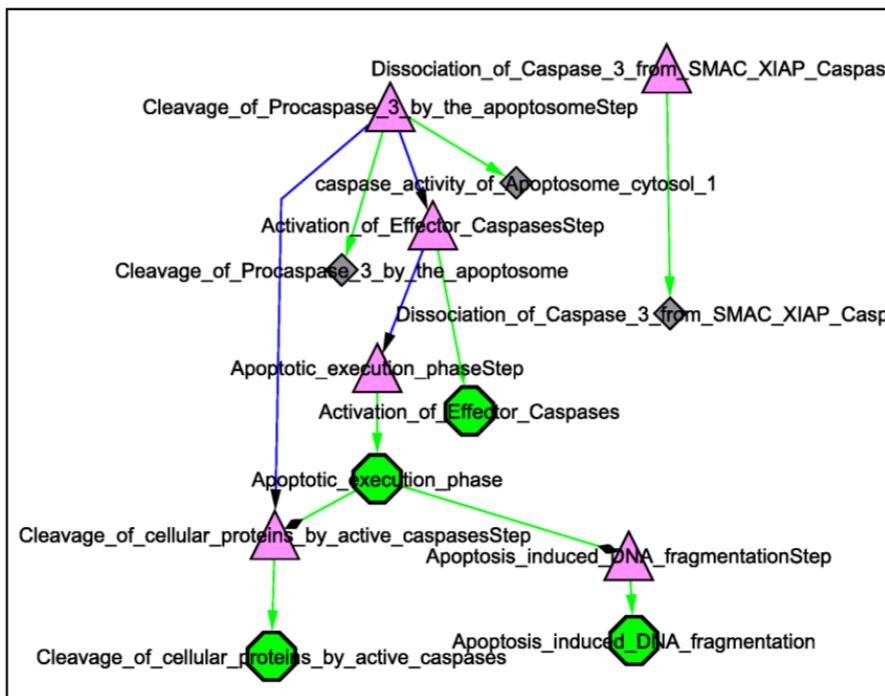


Figure 57: Fragment of Apoptosis from Reactome as Pathway Structure.

#### 9.4.3 BioPAX interface as Protein-Protein Interaction

Protein-protein Interaction interface contains only entities (not chemical species) with edges of CONTAINS and physicalInteraction type. This interface allows to visualize the composition of complexes like the Caspase3 example of the Apoptosis pathway (left, first), or, explicit information about protein interaction with TGFB1 (left, second), as in the NetPath TGF-beta BioPAX file. See figure ??.

#### 9.5 AIN file format

The AIN format describes a list of influences between genes, proteins, modified proteins or families. It is a table in ASCII, where the columns are separated by one tabulation (<Tab>).

The first line must start with the name of each column as follows (the titles are fixed):  
 ReviewRef ExperimentRef Link ChemType Delay Confidence Tissue Comment  
 (each space corresponds to a <Tab> on your keyboard).

- For the references (ReviewRef and ExperimentRef), if one wants to include a PUBMED number, it should have the form PMID:123456.
- The Link column describes a connection (activation or inhibition) between two entities, like A->B or A|B. The entities can be simply the name of a gene or a

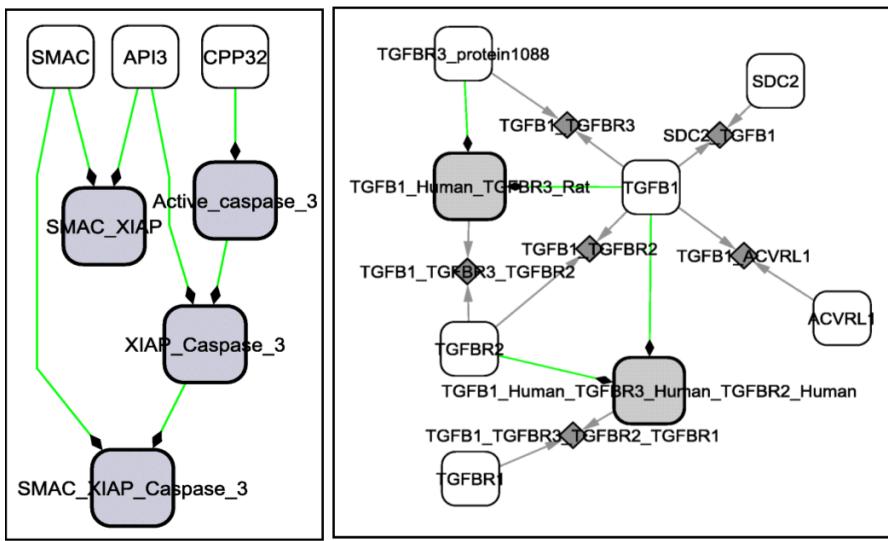


Figure 58: Fragment of Apoptosis from Reactome as Protein Protein Interaction.

protein, but it can also be a complex ((C:D)), a phosphorylated protein ((C^p)) or a family. In the latter case, the family can be given explicitly by the list of all its members ((C1,C2,C3)) or implicitly, by un undefined name ((C.)), where the . can be replaced by any character..

- In the other columns, if the user wishes to add more than one word in each field, the sentences need to be inserted between .
- If a field cannot be filled, a simple dot should be inserted.
- A # in first column makes the line comment.

For an example of AIN format, one can open the file ExamplApop.txt in a simple text editor or in spreadsheet as EXCEL. All the information in this AIN file is translated in BioPAX format when the file is imported in Cytoscape via BiNoM.

## 9.6 Modularization by shortest path clustering

When only the structure of a network is known, the simplest method to agglomerate nodes in a network is to put the closest nodes together. And so modules may have the fewest links between them. This method can lead to an algorithm of modularization of an oriented network. The notion of closeness and the process of creating modules are to be clarified.

The distance between nodes is based on the length of the shortest paths and the number of occurrences if several paths are equal (the equality of the shortest path is frequent in a strongly connected network). The distance from node 1 to node 2 is generally different

from distance from node 2 to node 1.

The used distance is the minimal linkage applied to the base distance, which is necessary to respect the triangular inequality. The distance between A and B is the minimum of distances from nodes in A to nodes in B and from nodes in B to nodes in A. And so, the agglomerative hierarchical clustering can be applied to build modules.

To avoid too speed increasing of clusters, they are ranked in a queue and the last created cluster is put at the end of the queue. For the same reason, nodes are sorted by in degree (sources in first). Despite of these precautions, the algorithm applied to strongly connected network gives unbalanced clusters (often a huge cluster and several tiny clusters). So, a ceiling number of nodes in a cluster must be fixed.

The agglomerative clustering gives 1 cluster at the end, which has no interest. Thats why; these 2 stop conditions are added:

- The length of the shortest path between 2 clusters reaches the maximal length.
- The number of clusters to be compared in the queue is less than 2.

The first stop condition make that too far clusters are not merged. When the last cluster to be created contains more than the maximal number of nodes, the largest cluster is excluded from the queue. Only the clusters remaining in the queue are to be compared by distance and they must be 2 or more.

The next page shows 3 examples (network inspired by toynet ). If the maximal length of the shortest paths is 1, nodes inside clusters are connected as a clique in a not oriented graph. But, if not, it may not be the case.

From a **practical point of view**, the input of ceiling number of nodes and maximal length of the shortest paths gives a set of not intersecting sub-networks. They are a partition of the network; their union is the whole network. This process is only useful for connected networks. Obviously, isolated nodes or sub-networks are not merged unless the maximal distance is infinity.

## 9.7 GLOSSARY

### 9.7.1 BioPAX

BioPAX is an OWL (Web Ontology Language) document designed to exchange biological pathways. BioPAX format provides separate layers of information: information about the reactions involved in the networks along with the participants, information about the structure of the pathway, and information about the protein-protein interactions.

### **9.7.2 CellDesigner**

CellDesigner is a structured diagram editor for drawing gene-regulatory and biochemical networks. Networks are drawn based on the process diagram, with graphical notation system proposed by Kitano.

### **9.7.3 BiNoM Index**

Directed labeled graph representing the objects in CellDesigner and BioPAX ontologies and their connections. Index maps only the information needed to display it and to identify the relevant information in the original CellDesigner or BioPAX files.

### **9.7.4 BiNoM interface**

Part of the BiNoM index (subgraph) visually presented by Cytoscape network. There are standard interfaces (Reaction network, pathway structure, protein interaction) which can be combined to construct a user-defined interface.

### **9.7.5 Optimal / suboptimal shortest paths**

Shortest paths in weighted directed graph paths in the graph between source and target nodes with minimal sum of weights of the edges making the path. Suboptimal path is constructed by removing all edges in all shortest paths one by one and one at a time and finding the shortest path.

### **9.7.6 Strongly Connected Components (SCC)**

A subgraph in a directed graph, in which there is path from any node to any node

### **9.7.7 Relevant cycle**

Any cyclic path in the graph which can not be decomposed further into simpler cycles

### **9.7.8 SBML**

Systems Biology Markup Language (SBML) is a standard for representing models of biochemical and gene-regulatory networks.