A key problem for many molecular biologists is the identification of methods like the majority vote algorithm (Schwikowski et al., 2001) and the network-based methods, although a comprehensive comparison was not performed. Indications that network-based methods outperform module-based methods, the data, for instance gene expression datasets (or direct methods) and module-based methods. In module-based methods, the data, for instance gene expression datasets or interaction networks, are clustered into modules which are functionally annotated using Gene Ontology (GO; Ashburner et al., 2000) or another functional categorization scheme. The functional annotation of a module is then transferred to its member genes. Some evidence however indicates that the module-based approach to predicting gene function may not be optimal. Wu et al. (2002) for example established that simply taking the top-10 correlated expression partners to predict the function of genes works better than traditional clustering methods. Similarly, Sharan et al. (2007) found indications that network-based methods outperform module-based methods, although a comprehensive comparison was not performed. Among the network-based methods, simple first-neighbor based methods like the majority vote algorithm (Schwikowski et al., 2000), where the function of a gene is predicted to be the most frequently occurring function among the gene’s direct network neighbors, often yield surprisingly good results compared with more sophisticated methods involving propagation of functional information through the network (Chua et al., 2006; Murali et al., 2006; Nabieva et al., 2005). Lossless propagation of functional annotations through the network, e.g. by considering the n-neighborhood of a gene with n > 1, generally gives rise to decreased performance, indicating that direct neighbors are the most relevant for predicting a gene’s function (Nabieva et al., 2005; Sharan et al., 2007). A range of network-based function prediction methods have been developed that employ more sophisticated machine learning techniques (Sharan et al., 2007 and references therein). These outperform naive methods but are computationally expensive, and because of this most of them have not been implemented as GUI-based tools (Sharan et al., 2007). A couple of web-based tools, for instance AraNet (Lee et al., 2009), ENDEAVOUR (Aerts et al., 2006) and GeneMania (Ward-Farley et al., 2010), allow users to prioritize candidate genes using probabilistic integrated networks. Although very useful, these tools are focused on one or a few organisms and, with the exception of GeneMania, do not allow users to upload their own datasets. Despite the importance of candidate gene discovery for molecular biology research, only a limited number of flexible tools have been developed for this purpose. Here we present PiNGO, a user-friendly tool that answers questions of the type: ‘Are there genes of class A in the network that are significantly connected to known class B genes but have no known role in process C?’ The main use for PiNGO is to screen networks for novel genes that could be involved in a particular process. For instance, if one would be interested in discovering novel transcriptional regulators of conjugation in yeast, the above sentence would read: ‘Are there transcriptional regulators in the network that are significantly connected to known conjugation genes but have no known role in conjugation?’ (see Figure 1). PiNGO is implemented as a plugin for Cytoscape, an open-source software platform to visualize, analyze and integrate molecular networks (Shannon et al., 2003).

1 METHODS AND IMPLEMENTATION

PiNGO implements a simple network-based method to find genes associated with processes or pathways of interest. The network to be analyzed can be loaded through Cytoscape or from a text file. Input networks may be gene coexpression networks, protein or genetic interaction networks, or integrated networks. Edge weights are not taken into account. Given one or more target GO categories, PiNGO screens the network for genes whose direct

© The Author 2011. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oup.com
neighbors are enriched for those functional categories at a chosen significance level. Subcategories of the target categories are also screened. PiNGO uses hypergeometric or binomial tests to calculate enrichment statistics, and Bonferroni or Benjamini–Hochberg FDR corrections to adjust the resulting $P$-values for multiple testing (Maere et al., 2005).

The results are summarized in the PiNGO output window. The candidate genes for each target category are listed along with $P$-values and associated raw counts that give a good indication of the prominence of the target category in the candidate gene’s neighborhood. When available, gene descriptions and known GO annotations of the candidate genes are reported to facilitate interpretation of the results and prioritization. In addition, an output network containing the candidate genes and their target GO annotated neighbors is extracted from the input network and displayed in Cytoscape. The output network reveals which genes contributed to the discovery of particular candidate genes. Output networks may contain denser areas or clusters of several candidate genes connected to the same neighbors. For Cytoscape input networks, all node and edge attributes and their visual descriptions and known GO annotations of the candidate genes are reported to facilitate interpretation of the results and prioritization. In addition, an output network containing the candidate genes and their target GO annotated neighbors is extracted from the input network and displayed in Cytoscape. The output network reveals which genes contributed to the discovery of particular candidate genes. Output networks may contain denser areas or clusters of several candidate genes connected to the same neighbors. For Cytoscape input networks, all node and edge attributes and their visual mappings are preserved in the output networks. For networks imported from text, the adjusted $P$-values of the candidate genes and the numbers of target GO annotated neighbors are mapped to the node color and size in the output network, respectively (see Figure 1).

A unique feature of PiNGO is the capability to exclude genes with certain functional properties from the analysis, or to focus on genes with particular functions. Users are typically not interested in rediscovering genes that are already known to be involved in the target process or closely associated processes. These processes can be excluded by specifying them as ‘filter categories’. People may also be interested primarily in discovering potential regulators of a given process, rather than effectors. In this case, categories like ‘transcription factor activity’ and ‘protein kinase activity’ can be specified as ‘start categories’, causing only genes annotated to these categories to be screened. Even when using filter categories, some not-so-novel genes may pop up in the candidate gene list, in the sense that their involvement in the chosen target process may already be known but not annotated in GO.

PiNGO offers unparalleled flexibility in the use of ontologies and annotations through its sister tool BiNGO (Maere et al., 2005), which has been refactored for this purpose and has to be installed separately. BiNGO provides default GO and GOSlim ontologies and annotations for a wide range of organisms, from bacteria to plants and animals. PiNGO also supports the use of custom ontologies, both in OBO format and flat text format, and annotations. This allows the user to use non-GO classification schemes or to use PiNGO on non-model organisms. Multiple identifier types and synonyms are supported when using built-in annotations or annotation files from the GO Consortium. PiNGO also features GO evidence code filtering, automated remapping of annotations to GOslim type sub-ontologies, and the possibility to specify custom reference gene sets against which functional enrichment is to be tested.

3 CONCLUSION
PiNGO implements a simple but efficient algorithm to find candidate genes in biological networks. PiNGO’s main strengths are its user-friendliness and flexibility.

ACKNOWLEDGEMENTS
We thank three anonymous reviewers and Thomas Van Parys for useful comments.

Funding: US National Institutes of Health (grant numbers RR031228 and GM070743 to M.S., K.O., and T.I.); Fund for Scientific Research-Flanders (FWO) to S.M.

Conflict of Interest: none declared.

REFERENCES