BIOINFORMATICS ORIGINAL PAPER Vol. 23 no. 22 2007, pages 3024–3031
doi:10.1093/bioinformatics/btm440

Gene expression

Improved detection of overrepresentation of Gene-Ontology annotations with parent–child analysis

Steffen Grossmann[1], Sebastian Bauer[2], Peter N. Robinson[2,*] and Martin Vingron[1,*]

[1]Max-Planck-Institute for Molecular Genetics, Ihnestrasse 73, 14195 Berlin and [2]Institute of Medical Genetics, Universitätsmedizin Charité, Augustenburger Platz 1, 13353 Berlin, Germany

Received on May 21, 2007; revised on August 3, 2007; accepted on August 20, 2007

Advance Access publication September 11, 2007

Associate Editor: Trey Ideker

ABSTRACT

Motivation: High-throughput experiments such as microarray hybridizations often yield long lists of genes found to share a certain characteristic such as differential expression. Exploring Gene Ontology (GO) annotations for such lists of genes has become a widespread practice to get first insights into the potential biological meaning of the experiment. The standard statistical approach to measuring overrepresentation of GO terms cannot cope with the dependencies resulting from the structure of GO because they analyze each term in isolation. Especially the fact that annotations are inherited from more specific descendant terms can result in certain types of false-positive results with potentially misleading biological interpretation, a phenomenon which we term the inheritance problem.

Results: We present here a novel approach to analysis of GO term overrepresentation that determines overrepresentation of terms in the context of annotations to the term's parents. This approach reduces the dependencies between the individual term's measurements, and thereby avoids producing false-positive results owing to the inheritance problem. ROC analysis using study sets with overrepresented GO terms showed a clear advantage for our approach over the standard algorithm with respect to the inheritance problem. Although there can be no gold standard for exploratory methods such as analysis of GO term overrepresentation, analysis of biological datasets suggests that our algorithm tends to identify the core GO terms that are most characteristic of the dataset being analyzed.

Availability: The Ontologizer can be found at the project homepage http://www.charite.de/ch/medgen/ontologizer

Contact: peter.robinson@charite.de and vingron@molgen.mpg.de

1 INTRODUCTION

High-throughput experiments such as microarray hybridizations often result in a list of genes (the *study set*) found to share a certain characteristic such as differential expression, and researchers are then confronted with the question of what differentiates the genes in the study set from the usually much larger set of all genes on a microarray chip (the *population set*). Exploring Gene Ontology annotations in this and in similar contexts has become a widespread practice to get first insights into the potential biological meaning of the experiment.

The Gene Ontology (GO) provides structured, controlled vocabularies and classifications for several domains of molecular and cellular biology (Ashburner *et al*., 2000). GO is structured into three domains, *molecular function*, *biological process* and *cellular component*. The terms of the GO form a directed acyclic graph (DAG), whereby individual terms are represented as nodes connected to more specific nodes by directed edges, such that each term is a more specific child of one or more parents. For instance, *mismatch repair* is a child of (more specific instance of) *DNA repair*. The Gene Ontology Annotation (GOA) Database and several other groups provide annotations for genes or gene products (hereafter simply referred to as genes) of over 50 species (Camon *et al*., 2004a). The *true-path rule* is a convention which states that whenever a gene is annotated to a term it is also implicitly associated with all the less specific parents of that term.

The most commonly used statistical test involves the hypergeometric distribution. This approach gives a straightforward and simple measure for the overrepresentation of an *individual* GO term, and we therefore use the term *term-for-term* approach to describe it (see Fig. 1 and Methods Section). It is applied to all terms individually and generally combined with some correction method for multiple testing to produce a list of terms which are accepted as being significantly overrepresented in the study set. A number of tools have been developed that implement a *term-for-term* analysis using the hypergeometric distribution or similar analyses, most of which are listed at the GO website (Gene Ontology Consortium, 2006).

The drawback of the *term-for-term* approach is that it does not respect dependencies between the GO terms that are caused by overlapping annotations. As a result of the *true-path rule*, each term in GO shares all the annotations of all of its descendants. A second source of overlapping annotations is that individual genes can be associated with multiple unrelated terms that are not connected in the GO DAG except by the root term.

In Alexa *et al*. (2006), two algorithms were presented which try to decorrelate the GO graph structure by processing the GO DAG in a bottom-up fashion, i.e. from most specific to least specific terms. In the first method, referred to as *elim*, the authors propose to eliminate the genes from the sets once

*To whom correspondence should be addressed.

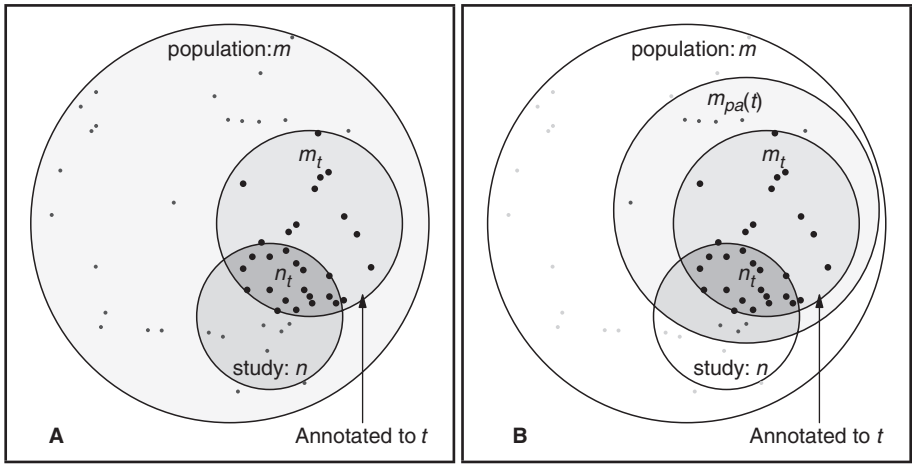

Fig. 1. Differences between *term-for-term* and *parent–child* analysis. Imagine that the genes are marbles of different colors in a jar. A marble is black if the corresponding gene is annotated to t, otherwise it is white. If we draw a certain number of marbles at random and without replacement from the jar we would expect the same proportion of white and black marbles among them as there was in the jar. We can calculate the probability of drawing a certain number of black marbles by chance using the hypergeometric distribution, whereby one sums over the upper tail of the distribution to obtain the probability of seeing at least a certain number of black marbles by chance. This approach is used in both the *term-for-term* and *parent–child* algorithms, though they differ in the definition of the sets that are analyzed as can be seen in the illustrations. **(A)** In *term-for-term* analysis, the probability is calculated of observing n_t or more genes annotated to t for a study set of size n given that m_t genes (depicted as bold dots) in the population of size m are annotated to t. **(B)** In *parent–child* analysis, we calculate the probability of observing n_t genes annotated to t in the study set given that $n_{pa(t)}$ genes in the study set are annotated to the parents of t ($n_{pa(t)}$ is given by the intersection of the study set with $m_{pa(t)}$). See the Methods section for further description.

they have been found to be associated with a GO term flagged as significant. Because this procedure can miss significant GO terms at less specific levels of the GO graph, the authors developed a second algorithm, which is referred to as *weight*. It examines connected nodes in the GO graph and down-weights genes that are annotated by less significant neighbors. A similar algorithm was implemented in the GOstats package of Bioconductor (Falcon and Gentleman, 2007).

We have developed a novel approach to statistical analysis of GO term overrepresentation that examines each term in the context of its parent terms, which we call the *parent–child approach*. A preliminary presentation of this approach was presented in a conference paper (Grossmann *et al.*, 2006). A related approach was mentioned as a part of a larger comparative analysis of yeast and bacterial protein interaction data in Sharan *et al.* (2005). However, algorithmic details were not given and a systematic comparison with the *term-for-term* approach was not carried out. Here, we develop two versions of the *parent–child* approach which we both compare systematically with each other and with the *term-for-term* approach as well as *elim* and *weight* to show their superiority over these approaches.

2 METHODS

2.1 Background: the *term-for-term* analysis

Denote the *population set* as P and the *study set* as S with sizes of m and n, respectively. Suppose that the term for which we want to measure overrepresentation is t. Let P_t be the set of genes annotated to t with cardinality m_t. S_t and n_t are analogously defined for genes in the study set S that are annotated to t. The situation is depicted in Figure 1A.

Suppose now that Σ is a set of size n sampled randomly without replacement from P, and let σ_t be the number of genes in Σ that are annotated to term t. The probability of observing exactly σ_t annotations can then be calculated according to the hypergeometric distribution:

$$\mathbb{P}(\sigma_t = k) = \frac{\binom{m_t}{k}\binom{m-m_t}{n-k}}{\binom{m}{n}}, \tag{1}$$

where, in general, $\binom{m}{n} = \frac{m!}{n!(m-n)!}$ is the number of ways of choosing a set containing n distinct elements out of a set of size m.

As we are interested in knowing the probability of seeing n_t or more annotated genes, we sum the term in (1) from n_t to the maximum possible number of annotations. This is equivalent to a one-sided Fisher exact test:

$$p_t(S) := \mathbb{P}(\sigma_t \geq n_t) = \sum_{k=n_t}^{\min(m_t,n)} \frac{\binom{m_t}{k}\binom{m-m_t}{n-k}}{\binom{m}{n}}. \tag{2}$$

2.2 The *parent–child* approaches

Denote by $pa(t)$ the parents of t and for simplicity suppose first that t has only a single parent. The probabilities involved the *parent–child* approaches are very similar to the one calculated in (1) except for conditioning on the event that the overlap of the random set Σ with $P_{pa}(t)$ is exactly as observed in the study set. From the *true-path rule* it follows that $m_t \leq m_{pa(t)}$ and the setting is illustrated in Figure 1B. Now, the probability of there being exactly σ_t annotations is:

$$\mathbb{P}(\sigma_t = k | \sigma_{pa(t)} = n_{pa(t)}) = \frac{\binom{m_t}{k}\binom{m_{pa(t)}-m_t}{n_{pa(t)}-k}}{\binom{m_{pa(t)}}{n_{pa(t)}}}. \tag{3}$$

To calculate significance, we sum over the probabilities for seeing $n_{pa(t)}$ or more annotations up to min (m_t, $n_{pa(t)}$) in an analogous manner to Equation (2).

If term t has more than one parent, it is not immediately apparent how to calculate the conditional probability in 3 (Grossmann *et al.*, 2006). We have chosen to examine in detail two approaches which lead

to solutions with a similar formal and computational complexity as the single-parent solution.

For the first approach, which we call *parent–child-union*, we define the sets of parents of a term t in the population and study set as the *union* of genes annotated the parents of t:

$$P^{\cup}_{pa(t)} := \bigcup_{u \in pa(t)} P_u, \qquad S^{\cup}_{pa(t)} := S \cap P^{\cup}_{pa(t)}$$

Therefore, we let $m_{pa(t)}$ and $n_{pa(t)}$ be the number of genes annotated to *any* of the parents of the respective sets.

For the second approach, which we call *parent–child-intersection*, we define the sets of parents of a term t as the *intersection* of genes that are annotated to the parents of t:

$$P^{\cap}_{pa(t)} := \bigcap_{u \in pa(t)} P_u, \qquad S^{\cap}_{pa(t)} := S \cap P^{\cap}_{pa(t)}$$

Hence, we count the number of genes annotated to *all* of the parents.

2.3 The *elim* and *weight* approaches

Both approaches were implemented as described in Alexa *et al*. (2006) except that we left out the direct Bonferroni adjustment of the *elim* method. For the *weight* method we chose $sigRatio(a, b) = \frac{a}{b}$ for weighting a parent–child node pair.

2.4 Gene Ontology terms and associations

Definitions of GO terms and associations between genes and GO terms were downloaded from the Gene Ontology consortium (Ashburner *et al*., 2000) website at `http://www.geneontology.org/`. The associations for yeast and human used in this article were provided by the *Saccharomyces Genome Database* (Dwight *et al*., 2002) and by the EBI (Camon *et al*., 2004b). The data were downloaded on 26 June 2007 and comprised of 4449 annotated terms.

2.5 All-subset minimal *P*-values

Unlike the *term-for-term* approach, the *parent–child* approaches capture a relative overrepresentation. Here, we introduce a measure which quantifies how well this is possible for a given term t. This *all-subset minimal P-value* is defined as

$$P_{\min}(t) := \min_{S \subseteq P} p_t(S) = p_t(P_t)$$

and marks the least P-value we can get by any possible study set. This set conforms to the study set made up exactly of all terms in the population that are annotated to t, or P_t. We will use the measure to filter out terms for which it is impossible to get a significant P-value regardless of the study set. For instance, if the annotations of a term t match those of its parent exactly, then $p_{\min}(t) = 1$.

2.6 Constructing artificial study sets

Artificial study sets with overrepresentation of a single term t were generated by sampling without replacement a *term percentage* $f_{\text{term}}(t)\%$ of genes annotated to t from the population together with a *noise percentage* $f_{\text{noise}}\%$ of genes not annotated to t. The population set P consisted of all yeast genes annotated to at least a single term.

To create a study set with overrepresentation of two terms t_1 and t_2 from one of the three subontologies with $f_{\text{term}}(t_1)$ and $f_{\text{term}}(t_2)$ and one *noise percentage* f_{noise} parameter, $f_{\text{term}}(t_1)$ percent of the genes are sampled from P_{t_1} as above. It is possible that some genes annotated to t_2 are also annotated to t_1 and have already been included in the study set. Therefore, genes are sampled from P_{t_2} only as necessary to obtain $f_{\text{term}}(t_2)$ percent of genes. Finally, f_{noise} percent genes are sampled from $P \backslash (P_{t_1} \cup P_{t_2})$, i.e. genes that are not annotated to either of the terms.

In this work, study sets were created as described using the population set of all genes in *Saccharomyces cerevisiae* that are annotated to at least one GO term. For some experiments, study sets were created only for terms for which the *all-subset minimal P-values* for both *parent–child* approaches are below 1×10^{-7}, in order to consider only terms for which it is possible to detect an overrepresentation with the methods under evaluation. This resulted in a total of 1115 different terms.

2.7 Dataset

The analysis of differential expression in the data derived from Kunikata *et al*. (2005) was performed using the limma package (Smyth, 2004) of Bioconductor (Gentleman *et al*., 2004). The set of all differentially expressed genes was used as the study set, and the population set was taken to be all genes represented on the microarray.

3 RESULTS

3.1 The inheritance problem of the *term-for-term* approach

We start by showing that the *term-for-term* approach is flawed because it does not take dependencies between parent and child terms into account. To do so, an artificial study set was constructed from *S.cerevisiae* data by overrepresenting the term *DNA repair* (GO:0006281) by including $f_{\text{term}}(t) = 50\%$ of all genes annotated to the term and $f_{\text{noise}}(t) = 10\%$ of the remaining terms in the population. We calculated the *term-for-term P*-value for each term and corrected for multiple testing with the resampling-based Westfall–Young approach (Westfall and Young, 1993) using 1000 resamplings. As expected, the term *DNA repair* itself is significantly overrepresented. A number of other terms are also flagged as significantly overrepresented including three children of *DNA repair* (Fig. 2). This is particularly surprising because it implies that there is more specific information in the dataset than has been put into it by means of its construction.

Observe that this also implies that the other eight children of *DNA repair* are not interesting for the study set. Both statements are not supported by the data in Table 1. We claim that this is an undesired effect that is caused by the fact that the *term-for-term* approach ignores the structure of the GO DAG.

This problem is of importance for researchers using such an analysis to explore the results of microarray or similar experiments. Given the results of the above example, a researcher might be tempted to examine *recombinational repair* specifically and neglect *postreplication repair* and the other children of *DNA repair* that were not flagged as significant. We consider this behavior of the *term-for-term P*-values to be a major drawback of the method and will refer to it as the *inheritance problem*.

3.2 Parent–child analysis outperforms term-for-term analysis with respect to the inheritance problem

The *parent–child* methods measure overrepresentation of a term t in the context of annotations to the parents of the term. Figure 1B presents an intuitive explanation of the approaches. We have examined two versions of algorithm which we call *parent–child-union* and *parent–child-intersection*. Details are provided in the Methods section.

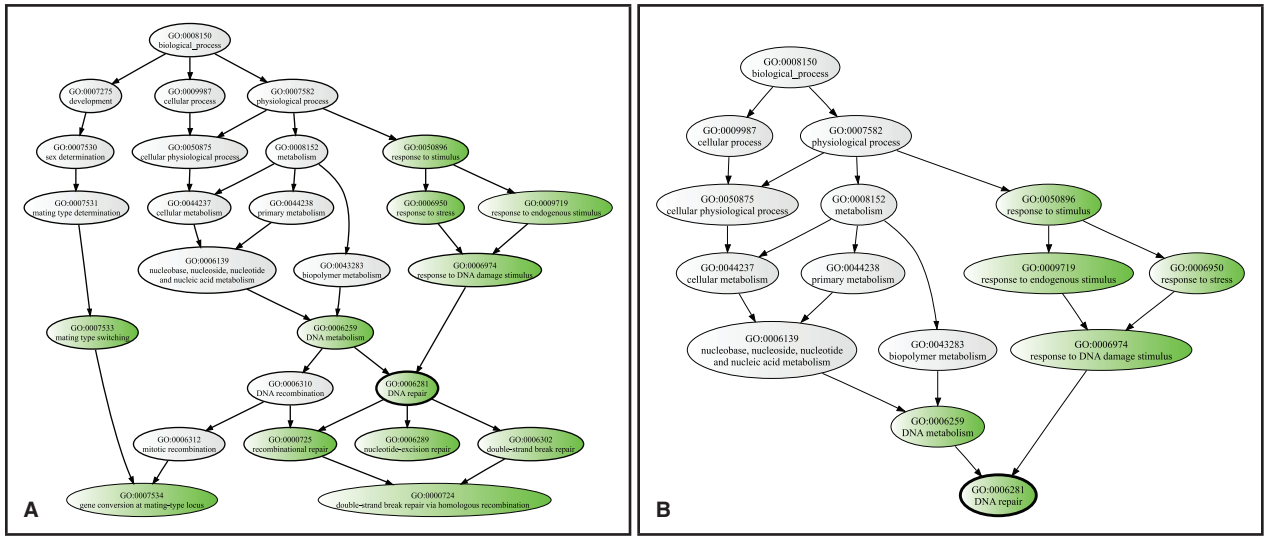

Fig. 2. Artificial overrepresentation of the GO term *DNA repair*. This term belongs to the *biological process* subontology, and we therefore restricted the analysis to terms in this subontology. (**A**) *Term-for-term* analysis. A subset of the GO graph with the significantly overrepresented terms and all their less specific parents is shown. Significantly overrepresented terms are highlighted in green. A total of 12 terms had a corrected *P*-value below the significance level of 0.05. (**B**) *Parent–child-intersection* analysis. None of the descendants of *DNA repair* are flagged as significant.

Table 1. Detailed data for the children of the term *DNA repair* from the analysis of the artificial study set

Term ID	Term name	m_t	n_t	p_t	n_t/m_t (%)
GO:0006302	Double-strand break repair	41	19	0.0 (*)	46.3
GO:0006289	Nucleotide-excision repair	31	15	0.001 (*)	48.4
GO:0000725	Recombinational repair	19	10	0.005 (*)	52.6
GO:0006298	Mismatch repair	20	9	0.077	45
GO:0000726	Non-recombinational repair	24	9	0.261	37.5
GO:0006307	DNA dealkylation	3	3	0.599	100
GO:0006284	Base-excision repair	10	3	1.0	30
GO:0019985	Bypass DNA synthesis	1	1	1.0	100
GO:0045021	Error-free DNA repair	4	2	1.0	50
GO:0006301	Postreplication repair	11	4	1.0	36.4
GO:0006290	Pyrimidine dimer repair	1	1	1.0	100

Notation: m_t: number of genes associated with the term in the population set; n_t: number of genes associated with the term in the study set; p_t: corrected *term-for-term P*-value. *P*-values with stars (*) are significant ($p_t < 0.05$). Terms are ordered by increasing *p*-values.

For the following experiments, we created 1115 study sets for *S.cerevisiae* genes in which one GO term was overrepresented as described in the Methods section. The study sets were analyzed with all methods to get the raw *P*-values, which were used to perform *receiver operator characteristics* (ROC) analysis for all combinations of term percentages of 75, 50 and 25% and noise percentages of 10 and 20%. In all settings, the *parent–child* approaches outperform the *term-for-term* approach. Moreover, the *parent–child-intersection* approach gives better results than the *parent–child-union* approach (Fig. 3).

All approaches loose their power with increasing noise percentage and decreasing term percentage. At the one extreme of a very weak signal, where $f_{\text{term}}(t) = 25\%$ and $f_{\text{noise}} = 20\%$ the *term-for-term* approach hardly performs better than the random method, whereas the *parent–child* approaches still have some ability to detect the overrepresented terms. With $f_{\text{term}}(t) = 75\%$ and $f_{\text{noise}} = 10\%$ the *parent–child-intersection* approach perfectly separates the overrepresented terms from their subterms. The performance advantage of the *parent–child* methods is similar when two terms are simultaneously overrepresented (Fig. 3).

The *parent–child* algorithms were designed especially to avoid false-positive results related to the inheritance problem, and the results presented above clearly demonstrate that they are superior to the *term-for-term* and the *elim* or *weight* algorithms in this regard. We note however that each of the three methods interrogates conceptually different measures of the significance of overrepresentation, and it is unclear whether a comparison such as that presented in Figure 3 is a fair comparison of the different methods. We therefore performed a similar ROC analysis using different subsets of GO terms with and without the restriction to terms satisfying a $p_{\min}$ value of 10^{-7}. When all terms are taken into consideration (i.e. not just the descendents of the overrepresented term), then the *weight* algorithm is superior to the *parent–child* algorithms for some but not all of the combinations of term and noise percentage in this setting, however, both algorithms are inferior to the *term-for-term* approach (Table 2). If all terms in the entire GO graph are considered that achieve a $p_{\min}$ value of 10^{-7} or better, then the *parent–child* methods are superior for all combinations tested.

3.3 Performance of the parent–child and term-for-term methods under multiple testing corrections

We next compared the performance of the *term-for-term* and *parent–child* procedures using multiple testing correction. We did not use ROC analysis because *P*-values that are nominally corrected to values more than 1 are truncated to 1. Moreover, the study sets have different sizes, resulting in

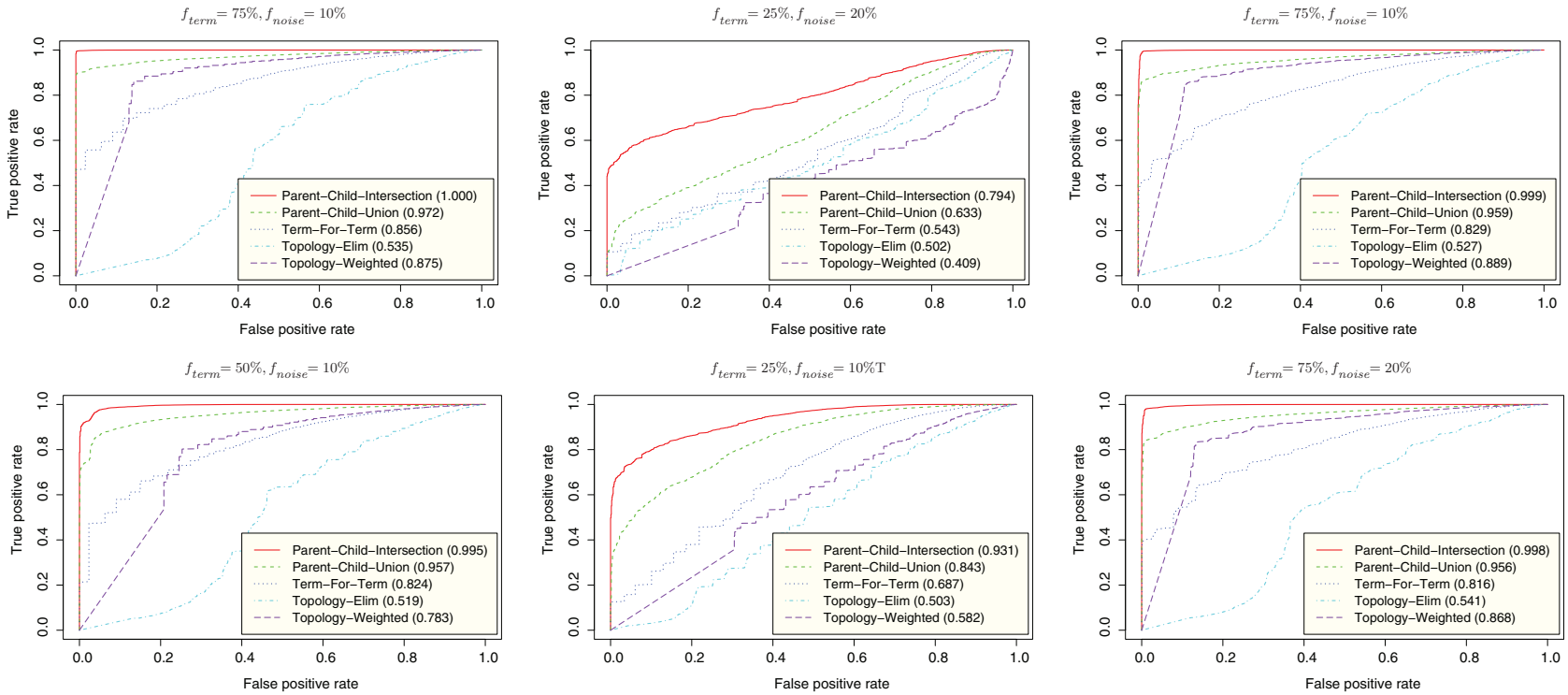

Fig. 3. ROC analysis of GO-term overrepresentation. The ROC curve plots the true positive rate of detecting the overrepresented term as significant as a function of the false-positive rate by which descendants of the term are flagged as significant, for varying *P*-value thresholds $c \in [0, 1]$. A perfect classifier results in a higher significance for t than for all other terms and receives a ROC score of 1.0. A random classifier would receive a score of ~0.5. The first two columns show results for single-term overrepresentation at the specified *term percentages* and *noise percentages*. For the third column, a total of 1000 different pairs of terms from the same subontology with a $p_{\min}$ below 1×10^{-7} for both *parent–child* approaches were sampled to construct the study sets. The performance of the different methods is comparable to the single-term case. Results for all combinations of term percentage from 25 to 90% and noise percentage from 5 to 25% showed a similar advantage for the *parent–child* algorithm (data not shown).

Table 2. ROC scores for selected studies

Setting	TfT	PCU	PCI	Elim	Weight	TfT	PCU	PCI	Elim	Weight	TfT	PCU	PCI	Elim	Weight
1/75/10	**0.953**	0.873	0.690	0.748	0.883	0.695	0.849	0.748	0.560	**0.850**	0.992	0.996	**0.997**	0.676	0.885
1/50/10	**0.916**	0.829	0.662	0.734	0.808	0.655	**0.804**	0.717	0.544	0.757	0.981	0.983	**0.987**	0.653	0.809
1/25/10	**0.790**	0.724	0.608	0.659	0.649	0.587	**0.702**	0.665	0.494	0.576	0.871	0.866	**0.900**	0.656	0.645
1/25/20	0.528	**0.529**	0.515	0.508	0.470	0.457	0.523	**0.588**	0.439	0.422	0.622	0.639	**0.733**	0.578	0.456
2/75/10	**0.943**	0.887	0.707	0.722	0.882	0.621	0.822	0.770	0.491	**0.861**	0.985	0.988	**0.991**	0.675	0.890
2/75/20	**0.920**	0.863	0.669	0.735	0.834	0.602	0.799	0.735	0.497	**0.828**	0.983	0.984	**0.988**	0.694	0.868
5/75/10	**0.922**	0.877	0.684	0.704	0.852	0.584	0.798	0.760	0.476	**0.868**	0.965	0.964	**0.972**	0.695	0.851

The left part lists the results of studies performed on all terms whereas the middle part shows the results when only the subterms of enriched terms are considered. The right part shows the result when considering only those terms with a $p_{\min}$ below 10^{-7}. The *Setting* column describes the settings of the artificial study set construction. Here, the first number represents the number of overrepresented terms, the second number the term percentage and the last number the noise percentage. The best ROC score for a given combination of settings is shown in bold for each of the three testing scenarios described above.

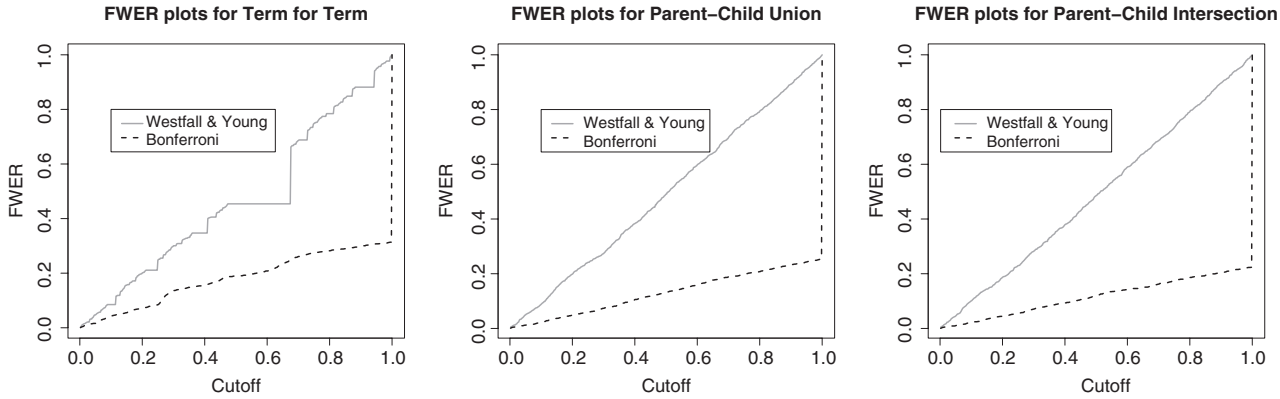

Fig. 4. Family-wise error rate plots. Total 2000 random study sets were generated and analyzed with each of the three methods. For any *P*-value cutoff $c \in [0, 1]$, the FWER can be estimated as the fraction of terms having a corrected *P*-value below *c*. Exact control of the FWER under is given when the resulting curve follows the main diagonal. Curves below the main diagonal indicate a too conservative procedure, curves above the main diagonal indicate that FWER control is not given for the procedure. In each plot, correction by the Bonferroni and Westfall–Young methods are compared.

different *P*-value correction ranges. Instead, we generated 2000 completely random study sets of size 250 and analyzed them with all three approaches each combined with two procedures for multiple testing corrections which control the FWER, the Bonferroni correction and the resampling-based Westfall–Young correction (Westfall and Young, 1993) with 5000 resamplings (Fig. 4).

As expected, the plots show that the Bonferroni procedure is much too conservative for all approaches. The best performance is given by the Westfall–Young correction in combination with either of the *parent–child* approaches. Here, exact control of the FWER is given. Interestingly, there seem to be some discretization effects when combining the Westfall–Young correction with the *term-for-term* approach. This can be explained by the fact that there are a large number of terms with equal *P*-values in the random study sets.

3.4 A biological example

In the following, we present an analysis of a dataset resulting from an experiments on the role of the prostaglandin E3 receptor EP3 (Kunikata *et al.*, 2005), in which saline (control)-challenged mice were compared against mice exposed to ovalbumin to induce asthma. We identified 246 differentially regulated genes with at least one GO annotation. Analysis with *parent–child-union* identified 17 overrepresented terms, analysis with *parent–child-intersection* identified 10 terms and analysis with *term-for-term* identified 63 terms. Figure 5 shows a portion of the graph emanating from *biological process*.

The term *immune response* has a total of nine children. The *term-for-term* approach identifies five of them as being significantly overrepresented, as well as numerous more distant descendants while *parent–child-union* identifies only *immune response* as being significantly overrepresented. This does not mean that the terms emanating from *immune response* are not important according to this analysis, merely that there is no statistical evidence to suggest that one particular descendent is more important than the others. The *parent–child-intersection* approach is generally more conservative than the *parent–child-union* approach. It identifies *physiological response to stimulus* as significant, which is a ancestor of *immune response*. Both *parent–child* methods identify other terms that characterize the dataset as being an allergic response including *MHC class II receptor activity*, *antigen binding* and *immunoglobulin complex*.

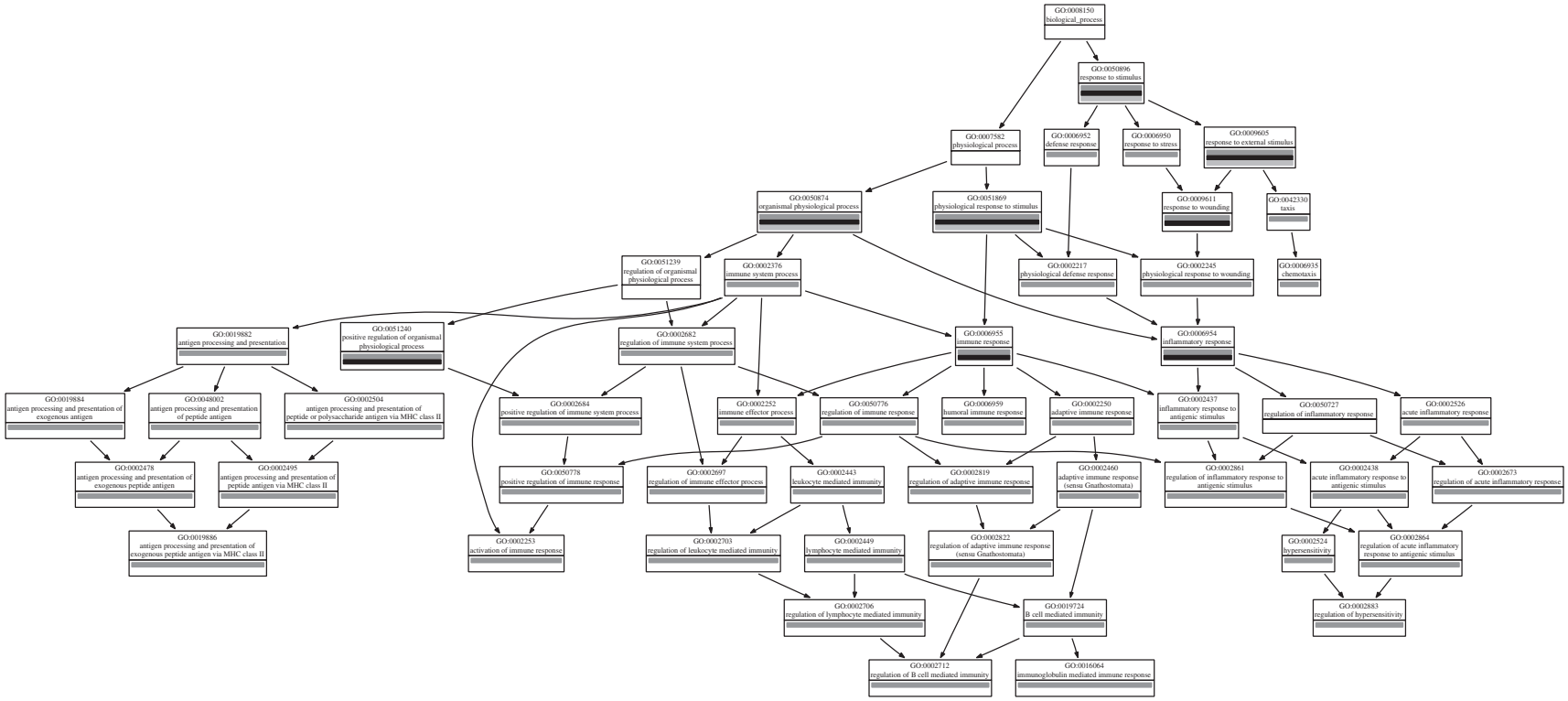

Fig. 5. Comparison of the three algorithms using real datasets. The study set is made up of genes differentially regulation between mice stimulated with ovalbumin to induce asthma and mice stimulated with saline (control). An excerpt of the GO graph is shown; each term has up to three bars denoting whether one of the three methods flagged the term as significantly overrepresented. The top bar represents the *term-for-term* approach, the middle bar represents the *parent–child-union* approach and the bottom bar represents *parent–child-intersection*. It is apparent that the *term-for-term* approach identifies many of the descendant terms of *response to external stimulus* and *immune response* as significant.

4 DISCUSSION

In this work, we have presented a novel algorithm for analysis of overrepresentation of annotations to GO terms. The *parent–child* procedure measures overrepresentation conditional on annotations to the parent of any term, whereas previous approaches measure overrepresentation of each term in isolation. We have shown that the *parent–child* procedure outperforms the standard procedure on two statistical measures.

A second phenomenon that differentiates *term-for-term* from *parent–child* analysis is that the *term-for-term* approach is able to pick up skewed distributions of annotations among the children of a given term. Although the amount of annotations to these terms might not be significant if analyzed in isolation, using the *parent–child* approaches, such skewed distributions may be identified. For instance, Liu *et al*. (2004) analyzed regulation of signaling genes by TGFβ during the *Caenorhabditis elegans* larval arrest stage (dauer). A number of genes showed regulation, including many hedgehog-related genes. The *parent–child-union* method, but not the *term-for-term* method, identified *hedgehog receptor activity* as significant, because 6/16 (38%) annotations of the parent term *transmembrane receptor activity* are inherited from the term *hedgehog receptor activity*, whereas in the population only 16 of the 864 annotations to *transmembrane receptor activity* are inherited from *hedgehog receptor activity* (6.4%).

The *parent–child* approaches conceptually measure the overrepresentation of terms in a different way than the *term-for-term* approach, and it is important to keep this in mind when interpreting results. In almost all datasets we have analyzed, the *parent–child* approaches identify a smaller number of terms as significantly overrepresented, and the *term-for-term* approach will flag many of the descendants of these terms as being overrepresented as well. Our results suggest that the *term-for-term* approach leads to false-positive results in these cases, in that the measured 'overrepresentation' results from the structure of the GO DAG and the number of annotated genes rather than truly reflecting the biology of the experiment at hand. There is an obvious danger for misleading interpretations of *term-for-term* analysis.

In contrast to the *elim*/*weight* approaches, the results of the *parent–child* approaches are derived from a single statistic for each term. Therefore, but also because the *parent–child* approaches' computational complexity matches the complexity of *term-for-term*, more sophisticated multiple test corrections which are based on permutations such as Westfall–Young can be applied easily.

The results of our ROC analysis of the *parent–child*, *term-for-term* and *elim*/*weight* algorithms showed that each of the methods has a performance advantage in certain testing scenarios. It was recently shown that the *elim*/*weight* methods have an advantage over the *term-for-term* approach among the top 150–615 significant genes Alexa *et al*. (2006). Given that the *parent–child* methods analyze a different measure of overrepresentation than the *term-for-term* and *elim*/*weight* methods, it is not clear which testing scenarios can be used to fairly compare the predicted accuracy of these methods on biological data. Our analysis clearly shows that the *parent–child* approaches are best able to cope with the inheritance problem. Further experience with newer methods such as the ones presented here and in Alexa *et al*. (2006) and Falcon and Gentleman (2007) will be required to estimate their usefulness for evaluating biological experiments.

5 CONCLUSION

There is no gold standard for the analysis of biological datasets for overrepresentation of GO terms, and any comparisons between methods are bound to be to some extent anecdotal. However, we have shown that the *term-for-term* approach can produce false-positive, and potentially biologically misleading results because it does not take the graph structure of GO into account. Our analysis using artificial datasets suggests that the *parent–child* approach avoids many of these problems. We provide an open-source Java implementation of the *term-for-term* and both *parent–child* algorithms within the framework of the Ontologizer at `http://www.charite.de/ch/medgen/ontologizer/`.

ACKNOWLEDGEMENTS

The research of S.B. and P.N.R. was supported by the SFB 760 grant of the *Deutsche Forschungsgemeinschaft* (DFG).

Conflicts of Interest: none declared.

REFERENCES

Alexa,A. *et al*. (2006) Improved scoring of functional groups from gene expression data by decorrelating GO graph structure. *Bioinformatics*, **22**, 1600–1607.

Ashburner,M. *et al*. (2000) Gene Ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet*., **25**, 25–29.

Camon,E. *et al*. (2004a) The Gene Ontology Annotation (GOA) Database an integrated resource of GO annotations to the UniProt Knowledgebase. *In Silico Biol*., **4**, 5–6, 1386-6338 Journal Article.

Camon,E. *et al*. (2004b) The Gene Ontology Annotation (GOA) Database: sharing knowledge in Uniprot with Gene Ontology. *Nucleic Acids Res*., **32**, D262–D266.

Dwight,S.S. *et al*. (2002) Saccharomyces Genome Database (SGD) provides secondary gene annotation using the Gene Ontology (GO). *Nucleic Acids Res*., **30**, 69–72.

Falcon,S. and Gentleman,R. (2007) Using GOstats to test gene lists for GO term association. *Bioinformatics*, **23**, 257–258.

Gene Ontology Consortium (2006) The Gene Ontology (GO) project in 2006. *Nucleic Acids Res*., **34**, D322–D326.

Gentleman,R.C. *et al*. (2004) Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol*., **5**, R80.

Grossmann,S. *et al*. (2006) An improved statistic for detecting over-represented Gene Ontology annotations in gene sets. In *Research in Computational Molecular Biology: 10th Annual International Conference, RECOMB 2006, Venice, Italy, April 2-5, 2006*, volume 3909 of *Lecture Notes in Computer Science*. pp. 85–98.

Kunikata,T. *et al*. (2005) Suppression of allergic inflammation by the Prostaglandin E receptor subtype EP3. *Nat. Immunol*., **6**, 524–531.

Liu,T. (2004) Regulation of signaling genes by TGFbeta during entry into dauer diapause in C. elegans. *BMC Dev. Biol*., **4**, 11.

Sharan,R. *et al*. (2005) Conserved patterns of protein interaction in multiple species. *Proc. Natl. Acad. Sci. USA*, **102**, 1974–1979.

Smyth,G.K. (2004) Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Stat. Appl. Genet. Mol. Biol*., **3**, Article 3.

Westfall,P. and Young,S. (1993) *Resampling-Based Multiple Testing*. Wiley, New York.