An Improved Statistic for Detecting Over-Represented Gene Ontology Annotations in Gene Sets

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

[1] Max Planck Institute for Molecular Genetics, Berlin, Germany
{steffen.grossmann, martin.vingron}@molgen.mpg.de
[2] Institute for Medical Genetics, Charité University Hospital, Humboldt University, Berlin, Germany
{sebastian.bauer, peter.robinson}@charite.de

Abstract. We propose an improved statistic for detecting over-represented Gene Ontology (GO) annotations in gene sets. While the current methods treats each term independently and hence ignores the structure of the GO hierarchy, our approach takes parent-child relationships into account. Over-representation of a term is measured with respect to the presence of its parental terms in the set. This resolves the problem that the standard approach tends to falsely detect an over-representation of more specific terms below terms known to be over-represented. To show this, we have generated gene sets in which single terms are artificially over-represented and compared the receiver operator characteristics of the two approaches on these sets. A comparison on a biological dataset further supports our method. Our approach comes at no additional computational complexity when compared to the standard approach. An implementation is available within the framework of the freely available Ontologizer application.

1 Introduction

The advent of high-throughput technologies such as microarray hybridization has resulted in the need to analyze large sets of genes with respect to their functional properties. One of the most basic approaches to do this is to use the large-scale functional annotation which is provided for several species by several groups in the context of the Gene Ontology (GO) ([1], [2], [3]).

The task is to detect GO terms that are over-represented in a given gene set. The standard statistic for this problem asks for each term whether it appears in the gene set at a significantly higher number than in a randomly drawn gene set of the same size. This approach has been discussed in many papers and has been implemented in numerous software tools ([4], [5], [6], [7], [8], [9], [10]). A p-value for this statistic can easily be calculated using the hypergeometric distribution. Since this approach analyzes each term individually, without respect to any relations to other terms, we refer to it as the *term-for-term* approach.

A. Apostolico et al. (Eds.): RECOMB 2006, LNBI 3909, pp. 85–98, 2006.

The term-for-term approach becomes problematic if one looks at several or all GO terms simultaneously.There are two properties of the GO annotation which result in a complicated dependency structure between the p-values calculated for the individual GO terms. First, the annotation is done in a hierarchical manner such that genes which are annotated to a given GO term are also implicitly annotated to all less specific terms in the hierarchy (the so-called *true path rule*). Second, individual genes can be annotated to multiple GO terms, which reside in very different parts of the GO hierarchy. Both properties have the effect that information about the over-representation of one GO term can carry a substantial amount of information about the over-representation of other GO terms. This effect is especially severe when looking at parent-child pairs. Knowing that a certain term is over-represented in many cases increases the chance that some of its descendant terms also appear to be over-represented. We call this the *inheritance problem* and we consider it to be the main drawback of the term-for-term approach.

In this paper, we propose a different statistic to measure the over-representation of individual GO terms in a gene set of interest. Our method resolves the inheritance problem by explicitly taking into account parent-child relationships between the GO terms. It does this by measuring the over-representation of a GO term *given* the presence of all its parental terms in the gene set. Again, p-values can be calculated using the hypergeometric distribution at no increased computational complexity. We call our approach the *parent-child* approach. A related approach was mentioned as a part of a larger comparative analysis of yeast and bacterial protein interaction data in [11]. However, algorithmic details were not given and a systematic comparison with the term-for-term approach was not carried out.

The rest of the paper is organized as follows. In Section 2 we first review the term-for-term approach and discuss the inheritance problem in more detail. The new parent-child approach is then explained and the rationale behind it is explained. Section 3 is devoted to a comparison of the parent-child approach with the term-for-term approach. We compare the two approaches on gene sets with an artificial over-representation of individual terms. This illustrates that the parent-child approach solves the inheritance problem. We finish the section by comparing the two methods on a biological dataset. The paper is closed by a discussion.

2 Method

Given a set of genes of interest we want to analyze the functional annotation of the genes in the set. A typical example of such an analysis involves a microarray experiment where the gene set would consist of the genes which are differentially expressed under some experimental condition. We will use the name *study set* for such a gene set in the following and denote it by S. We suppose that the study set appears as a subset of the larger set of all the genes which have been considered in the experiment (such as the set of all genes which are represented on the microarray). We will call this set the *population set* and denote it by P.

The functional annotation we want to analyze consists of an assignment of some of the genes in the population set to the terms in the GO. Individual genes can be annotated to multiple GO terms. The relations between the GO terms have the structure of a *directed acyclic graph* (DAG) $G = (T, H)$, where T is the set of GO terms and the relation $H \subset T \times T$ captures the parent-child relationships (i.e. we have $(t_1, t_2) \in H$ whenever t_1 is a direct parent of t_2). In this relationship the children correspond to the *more* specific and the parents to the *less* specific terms. The set of parents of a term t is denoted by $\mathrm{pa}(t)$. We also use ρ to denote the unique *root term* of GO which has no parents.

For any GO term t, we denote by P_t the set of genes in the population set that are annotated to this term. The convention is that the annotation of the children of t is also passed to t itself (the so-called *true path rule*). This has the effect that $P_{t'} \subseteq P_t$ whenever $t \in \mathrm{pa}(t')$. When we speak about the *directly assigned* genes of a term t we mean those genes which are assigned to t but not to any of its children. Observe that the population set might also contain genes for which no assignment to any GO term is given. This means that $P \backslash P_\rho$ might be non-empty. As a shorthand notation we will write $m_t := |P_t|$ to denote the size of the set P_t, and the size of the whole population set P will be denoted by m. For the study set we use a corresponding notation by writing S_t and defining $n_t := |S_t|$ and $n := |S|$.

2.1 The Term-for-Term Approach and the Inheritance Problem

The statistic used by the term-for-term approach to measure the over-representation of a GO term t is based on comparing the presence of the term in the study set to its presence in a randomly drawn subset from the population set of the same size as the study set. The over-representation is quantified by calculating the probability of seeing in such a randomly drawn set at least as many term-t genes as in the study set. Formally, let Σ be a set of size n which has been drawn randomly from P. We write $\sigma_t := |\Sigma_t|$ for the number of genes annotated to term t in this random set. The probability of interest can now be easily calculated as the upper tail of a hypergeometric distribution

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

Heuristically formulated, the *inheritance problem* lies in the fact that once it is known that a certain term t is over-represented in the study set there is also an increased chance that descendant terms of t get a small p-value and are also classified as over-represented. The reason for this clearly lies in the fact that the statistical test for each term is carried out in isolation, without taking annotations of other terms into account. The impact of this can be seen by the following thought experiment.

In a typical population set, annotation to GO terms is usually not available for all genes (meaning that $m_\rho < m$). Suppose term-for-term p-values are calculated with respect to a random sampling of a size-n set from the population set. It might be the case, that there is an unusually high number of genes from P_ρ in the study set, resulting in a very low $p_\rho(S)$-value. In such a case it should not be surprising that also the terms on the next levels below the root ρ have small p-values and therefore appear to be over-represented. However, *given* that $|\Sigma_\rho| = n_\rho$, this (artificial) over-representation should vanish. This effect has already been taken into account in as far as the analysis of all terms is often done with replacing S with S_ρ and P with P_ρ. Although this is already a reasonable modification, we claim that the same problem holds true for terms inside the GO-DAG. Suppose, e.g., that the study set contains a significantly high number of genes which are annotated to the term *metabolism*. Again, it should not be surprising to also see a significant over-representation for the more specific forms of *metabolism* which are represented by the children of the term *metabolism*.

These heuristic considerations motivated us to develop our new parent-child approach.

2.2 The Parent-Child Approach

The parent-child approach uses a statistic to detect over-represented terms, which compares the term's presence with the presence of its parental terms in the study set.

Let's first consider a term t which has a unique parent t' in the hierarchy of GO terms. The idea behind the parent-child approach is to compare the presence of term t in the study set to a random set of genes in which t' is present *at the same number* as in the original study set S. To quantify this, we draw a random subset of size $n_{t'}$ from P_t and calculate the probability $\hat{p}_t(S)$ to see at least n_t term-t genes in that set. Again, this can be done using the hypergeometric distribution and results in

$$\hat{p}_t(S) := \mathbb{P}(\sigma_t \geq n_t | \sigma_{t'} = n_{t'}) = \sum_{k=n_t}^{\min(n_{t'},m_t)} \frac{\binom{m_t}{k}\binom{m_{t'}-m_t}{n_{t'}-k}}{\binom{m_{t'}}{n_{t'}}}, \qquad (1)$$

where $(\sigma_t)_t \in T$ is again defined on a randomly drawn subset Σ of size n.

However, the assumption that a GO term has a single parent is not valid for the GO hierarchy, since it has the structure of a directed acyclic graph. Heuristically formulated, when there are several parents of a GO term t, we want to measure the over-representation of t, given the presence of *all its parents* in the study set. When trying to formalize this, we see that there are at least two ways to quantify the presence of the parents in the study set. Enumerate the parents of t as $\mathrm{pa}(t) = \{t_1, \dots, t_l\}$. The first idea would be to condition on the numbers $(n_{t_i})_{1 \leq i \leq l}$, i.e. to calculate the probability

$$\mathbb{P}_0(\sigma_t \geq n_t | \sigma_{t_1} = n_{t_1}, \dots, \sigma_{t_l} = n_{t_l}).$$

It turns out that it becomes extremely difficult to combine different hypergeometric weights to calculate this probability. The reason for this is that, for example, if we know the value of n_{t_1} and n_{t_2}, it is not clear in how far the genes annotated to t_1 overlap with those annotated to t_2. Because of the true path rule we know that both share at least a set of n_t genes. However, there are many potential ways of partitioning the genes in P_{t_1} or P_{t_2} but not in P_t among the other children of t_1 and t_2 or among direct annotations to these terms. This becomes even more complicated when considering combinations of more than two parents.

We therefore chose another generalization of (1). What we fix in the comparison of t to its parents is the total number of genes in the study set, which are annotated to any of the parents. A set of this size is randomly drawn from the collection of all genes in the population set which are annotated to any of the parents of t and the probability of seeing at least n_t term-t genes in such a set is calculated. To formalize this, define

$$n_{\mathrm{pa}(t)} := \left| \bigcup_{t' \in \mathrm{pa}(t)} S_{t'} \right|$$

and correspondingly $m_{\mathrm{pa}(t)}$ and $\sigma_{\mathrm{pa}(t)}$. Our final definition of parent-child p-values is

$$\hat{p}_t(S) := \mathbb{P}_0(\sigma_t \geq n_t | \sigma_{\mathrm{pa}(t)} = n_{\mathrm{pa}(t)}) = \sum_{k=n_t}^{\min(n_{\mathrm{pa}(t)}, m_t)} \frac{\binom{m_t}{k}\binom{m_{\mathrm{pa}(t)}-m_t}{n_{\mathrm{pa}(t)}-k}}{\binom{m_{\mathrm{pa}(t)}}{n_{\mathrm{pa}(t)}}}. \quad (2)$$

This definition simplifies to (1) when there is a unique parent of term t. The advantage of this definition is that it comes at no increased computational complexity when compared to the term-for-term approach.

2.3 Implementation in the Ontologizer

We have implemented the term-for-term approach and our new parent-child approach in Version 2.0 of the Ontologizer [9]. Executables and source code are available from the authors at `http://www.charite.de/ch/medgen/ontologizer` and can be used under the GNU public license.

Due to the importance of multiple testing corrections (MTCs) a selection of different approaches is also available in the Ontologizer. Both of the methods to calculate raw p-values can be combined with any of the implemented MTC approaches.

To produce the results in this paper we used both calculation methods in combination with the standard *Bonferroni* and the *step-down resampling* correction by Westfall & Young [12]. Both control the family-wise error rate. The resampling method is known to be less conservative.

The resampling needed in the step-down method is done by randomly selecting a gene set of the same size as the analyzed study set from the whole population set. This is the natural adaptation of the resampling strategy as described in [13] for resampling based MTCs in the context of microarray data analysis.

3 Comparing the Two Approaches

To compare our new parent-child approach with the term-for-term approach we developed a strategy which allows us to compare the respective false positive rates when over-representation of a certain term is given.

To this end, we generated gene sets in which a given GO term t is artificially over-represented. The most naive way to do this is to take the subset P_t of genes which are annotated to the term in the population set P. More realistic examples of such sets can be obtained by combining a certain proportion of genes from P_t with a certain amount of genes randomly drawn from P. When testing such sets for over-representation of GO terms, the term t itself should be detected along with some other terms which can then be considered as false positives.

The results presented in the next two subsections are based on a population set of 6456 yeast genes for which we downloaded about 32000 annotations to a total of 3870 different GO terms from the *Saccharomyces Genome Database* (`http://www.yeastgenome.org/`, version as of August 12th, 2005, [14]). We used the yeast annotation for no particular reason, results obtained with other species were comparable.

3.1 All-Subset Minimal *p*-Values

Suppose we are given a study set S for which we know that a certain term t is over-represented. To detect this, it is necessary that the p-value calculated under the respective method for that term t is small enough to remain significant even after correction for multiple testing.

Since the parent-child method measures over-representation of a term with respect to the presence of its parental terms in the study set, it can happen that there are terms for which it can be already seen from the population set P that any significant over-representation can not occur. This effect can be quantified by looking at what we call the *all-subset minimal p-value* $\hat{p}_t^{\min}$ of a term t. This is the minimal p-value one can obtain when minimizing the $\hat{p}_t(S)$ values over all possible study sets or, formally,

$$\hat{p}_t^{\min} := \min_{S \subseteq P} \hat{p}_t(S) = \hat{p}_t(P_t).$$

The claim of the last equation enables us to calculate the all-subset minimal p-values and can easily be checked using elementary probability theory. The corresponding statement is also true for the term-for-term approach, where we have $p_t^{\min} := \min_{S \subseteq P} p_t(S) = p_t(P_t)$. The behavior of the all-subset minimal p-values differs tremendously between the two approaches.

The histogram in Figure 1 a) shows that for the parent-child approach there is obviously a large number of terms for which the all-subset minimal p-values are *not small.* This can be explained by almost trivial parent-child relations which are already fixed by the annotations of the population set P. More explicitly, denote by $P_t \subseteq P_{\mathrm{pa}(t)} := \bigcup_{t' \in \mathrm{pa}(t)} P_{t'}$ the set of genes annotated to at least one of the parents of t. If there is no sufficiently large (set-)difference between P_t and

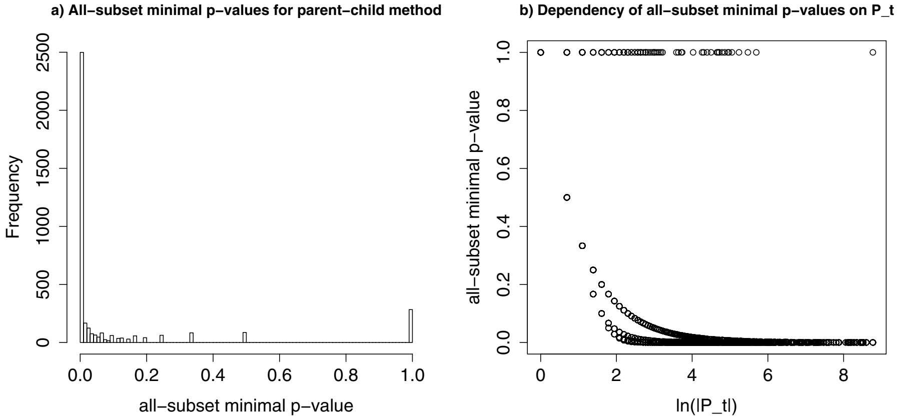

Fig. 1. a) The distribution of all-subset minimal p-values of the parent-child approach. It can be seen that there is a substantial amount of terms for which the all-subset minimal p-value is not small. In contrast, all-subset minimal p-values of the term-for-term approach are always small (below $1.55 \cdot 10^{-4}$ in this dataset). We discuss the reasons for this in the text. b) Scatterplot of the logarithm of the number of genes annotated to a term against the all-subset minimal p-values of the term. It can be seen that high minimal p-values are more likely to appear at terms with only a few genes annotated. The obvious arrangement in curves corresponds to cases where m_t and $m_{\mathrm{pa}(t)}$ differ only by 0, 1, 2,. . . annotated genes. The maximal value of m_t for which we observe a trivial $\hat{p}_t^{\min}$ value of one goes up to 297 ($\ln(297) \approx 5.7$) (the dot in the upper right corner corresponds to the root term, which is always trivial).

$P_{\mathrm{pa}(t)}$, the value of $\hat{p}_t^{\min}$ cannot be small. In the extreme cases where $\hat{p}_t^{\min} = 1$ we have $P_t = P_{\mathrm{pa}(t)}$.

From Figure 1 b), where we plot $\hat{p}_t^{\min}$ values against the corresponding $\ln(P_t)$ values for all terms t, it can be seen that large values mainly occur for those terms to which only few genes are annotated in the population set (cf. figure legend for more details).

In contrast to the parent-child approach, the term-for-term approach always produces extremely small all-subset minimal p-values. This is not surprising, because since $p_t(P_t)$ is the probability that a set of size $m_t := |P_t|$ drawn randomly from P consists exactly of the genes in P_t it should always be small. This is related to our criticism of the term-for-term approach. We criticize that once we know about the presence of a certain term in the study set, we also have some information about the presence of its descendant terms in the set. This knowledge is reflected by our parent-child approach but neglected by the term-for-term approach.

3.2 False Positive (Descendant) Terms

Our strategy to compare the two approaches with respect to the false positive prediction of over-represented GO terms is the following. We create a large number of (artificial) study sets for each of which a single term is intentionally over-represented. When analyzing such a set with one of the methods, any term found to be over-represented can be counted as a *false positive* classification, unless it is the intentionally over-represented term itself. We compare those false positive counts in terms of *receiver operator characteristics* (ROC) curves to visualize the differences between the two approaches. The technical details of this strategy need a more thorough description which we give now.

We start by selecting those terms which we will intentionally over-represent in the creation of the study sets. According to the results from the last subsection, we restrict ourself to those terms for which a statistically significant over-representation is possible. Therefore, we identified the set

$$T_{\text{good}} := \{t \in T \colon \hat{p}_t^{\min} < 10^{-7}\}$$

of terms with a small enough all-subset minimal p-value. We chose a cutoff of 10^{-7} because it leaves us enough room to get small p-values even after correction for multiple testing. In our concrete dataset, a total of 1472 out of 3870 terms made it into T_{good}.

For each term in $t \in T_{\text{good}}$ we construct artificial study sets at different levels of over-representation of t and different levels of *noise* as follows. We start with the set P_t from which we keep a certain proportion (called *term proportion*) in the study set by a random selection. To those genes we add another proportion (called *population proportion*) of genes from the whole population set as random noise. We did this for term proportions of 100%, 75% and 50% and population proportions ranging from 0% to 25% at steps of 5% resulting in a total of 18 parameter combinations.

Let S be a study set constructed as just described and let $t_{\text{over}}(S)$ be the term over-represented in the its construction. S is analyzed with both methods and the results are further processed to count the respective false positive and negative predictions. Observe that any analysis of S naturally divides the total set of terms T into two parts. First, there is the set of terms which do not annotate any of the genes in S. We do not consider those terms as true negatives in the calculation of the false positive rate, because both methods will never have a chance to falsely predict any of those terms as over-represented and therefore will agree. Moreover, we restrict ourselves to those terms which reside in the same of the subontologies (defined by the terms *biological process*, *molecular function* and *cellular component*) of GO as the term $t_{\text{over}}(S)$. The reason for this is that there are many biologically meaningful relations between terms in different subontologies which are also respected in the annotation of the genes. The set of terms which is left after this reduction will be considered in the calculation of true/false positives and be denoted by $T_{\text{test}}(S)$. By construction, any term in $T_{\text{test}}(S)$ other than $t_{\text{over}}(S)$ will be treated as a false positive when predicted as over-represented at a certain p-value cutoff by either method. The term $t_{\text{over}}(S)$ itself is counted as a false negative when not detected at that cutoff.

A last distinction has to be explained to understand the two final analyses we present. To better highlight the inheritance problem we first intersect T_{test} with the set of all *strict* descendant terms of $t_{\text{over}}(S)$ and count the false positives only on this set which we denote by $T_{\text{desc}}(S)$. In the second analysis, the counting is done on the whole of T_{test} to compare the general tendency to falsely classify terms as over-represented.

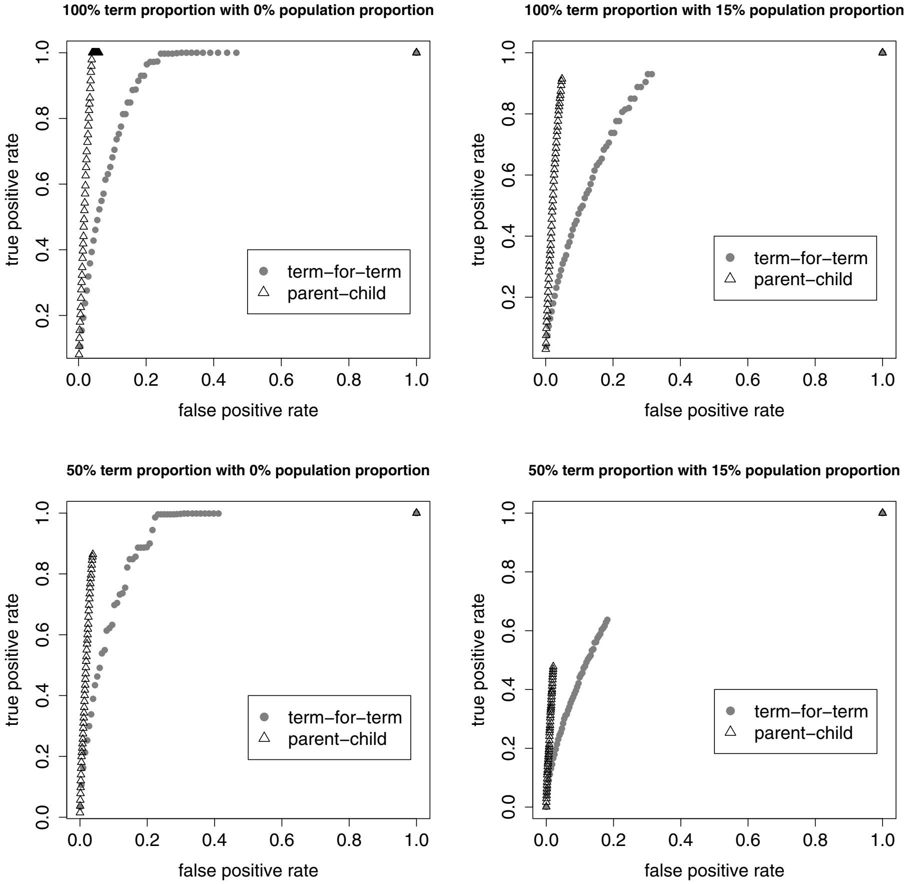

Fig. 2. Descendant-term ROC at different combinations of term and population proportions. Each point corresponds to the false and true positive rate calculated at a certain p-value cutoff π. We did not connect the points with lines, because those would indicate nonexisting combinations of the two rates. The parent-child method drastically reduces the number of descendant terms falsely predicted to be over-represented. Adding noise or reducing the level of over-representation makes it harder for both methods to correctly detect the over-represented term. This is the reason for the breaking off of the curves. ROC analysis of other combinations of term and population proportions always showed a clear advantage of the parent-child approach.

To calculate true/false positive rates, we combine the results from all study sets for a fixed combination of term and population proportions. Let $\mathcal{S}$ be such a collection of study sets.

We begin with the analysis where we count false positives on $T_{\text{desc}}(S)$ only. For a given p-value cutoff π we define the *descendant-term false positive rate* $\text{FPR}_{\text{desc}}(\pi)$ of the term-for-term method over the set $\mathcal{S}$ as

$$\text{FPR}_{\text{desc}}(\pi) := \frac{\sum_{S \in \mathcal{S}} \left|\{t \in T_{\text{desc}}(S) \colon p_t(S) < \pi\}\right|}{\sum_{S \in \mathcal{S}} |T_{\text{desc}}(S)|} \tag{3}$$

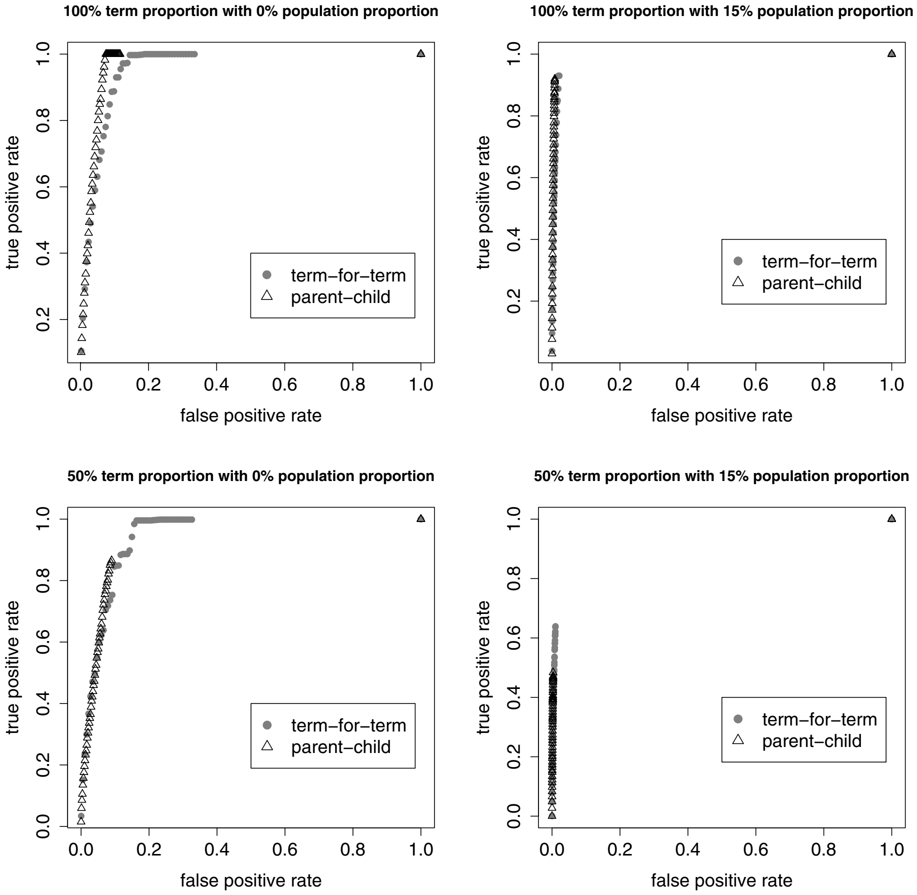

Fig. 3. All-term ROC at different combinations of term and population proportions. It can be seen that the parent-child method performs at least as well as the term-for-term method. Again, adding noise or reducing the level of over-representation has an impact on both method's ability to correctly detect the over-represented term. Additional remarks are in the legend to Figure 2.

and the *descendant-term true positive rate* $\mathrm{TPR}_{\mathrm{desc}}(\pi)$ as

$$\mathrm{TPR}_{\mathrm{desc}}(\pi) := \frac{\left|\{S \in \mathcal{S} : p_{t_{\mathrm{over}}(S)} < \pi\}\right|}{|\mathcal{S}|}. \tag{4}$$

The corresponding descendant-term false and true positive rates for the parent-child method are denoted by $\widehat{\mathrm{FPR}}_{\mathrm{desc}}(\pi)$ and $\widehat{\mathrm{TPR}}_{\mathrm{desc}}(\pi)$ and calculated by replacing p with $\hat{p}$ in (3) and (4).

A *receiver operator characteristics* (ROC) curve is obtained from those values by plotting the false positive rate versus the true positive rate for all p-value cutoffs π between 0 and 1. The results for the descendant-term analysis are shown in Figure 2 for some combinations of term and population proportions. It can be seen that the parent-child method drastically reduces the number of

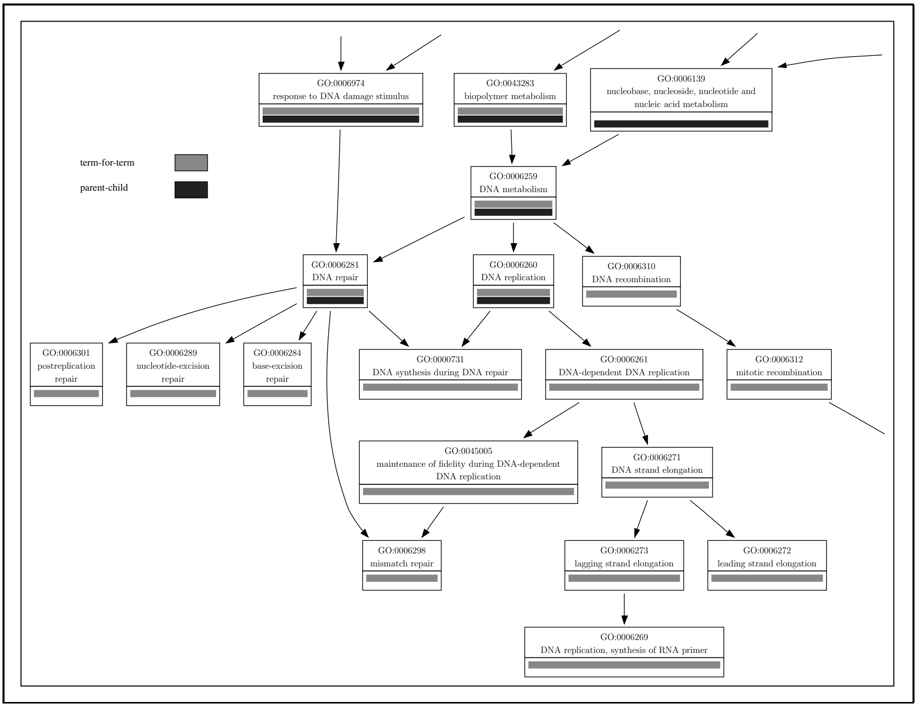

Fig. 4. Excerpt of the graph displaying over-represented terms in a set of 300 yeast genes shown to be specific for cell-cycle phase G1 ([15]). A gray marker below a term means that the term is over-represented in the term-for-term approach while a black marker indicates over-representation in the parent-child method. The inheritance problem of the term-for-term approach can be seen among the descendant terms of the two terms *DNA repair* and *DNA replication*. The range of the specific aspects of *DNA replication* and *DNA repair* found by the term-for-term approach is so wide, that no specific biological information can be gained from this. The figure was generated by post-processing output from the Ontologizer with the *Graphviz* program [16].

descendant terms falsely predicted to be over-represented when compared to the term-for-term approach.

In the second analyses we calculate the false positive rates using all terms in $T_{\text{test}}(S)$. The results in Figure 3 show that the parent-child approach performs comparably to the term-for-term approach with respect to this general counting of false positives.

3.3 A Biological Example

We compare the two approaches on a study set from a set of *Saccharomyces cerevisiae* cell cycle experiments, in which approximately 800 genes were shown to be cell-cycle regulated ([15]). We present an analysis of the 300 G1-specific genes, taking the entire set of genes on the microarray as population set. GO annotations from the *Saccharomyces Genome Database* ([14]) were used.

Although the G1 stage precedes the S, or synthesis, stage when DNA replication occurs, the G1 cluster contains many genes involved in DNA replication and repair, budding, chromatin and the spindle pole body (cf. Fig. 7 of [15]).

In Figure 4 we present a portion of the results of the GO analysis using both the parent-child and the standard term-for-term method. For both methods *p*-values were corrected by Westfall & Young's step-down resampling correction ([12]). We think that most of the terms which are identified by the term-for-term approach but not by the parent-child method are there because of the inheritance problem. According to the parent-child method, the key terms in this dataset are *DNA repair* and *DNA replication*. The descendant terms which are additionally identified by the term-for-term approach don't show a tendency towards a selection of closely related more specific terms, but rather cover a wide range of different terms. We don't claim that these more specific terms are biologically irrelevant. We only claim that there is no evidence that a certain collection of those terms plays an increased role in the study set.

4 Discussion

With the parent-child approach we have introduced a novel statistic to measure over-representation of GO terms in GO annotated gene sets. The motivation for this was the *inheritance problem* of the term-for-term approach which is the current standard. The inheritance problem refers to the fact that if a certain GO term is over-represented in a gene set, the term-for-term approach has a tendency to incorrectly show an over-representation of some of its descendant terms. We have illustrated this problem by analyzing gene sets in which we artificially introduced different levels of over-representation of individual terms. Analyzing the gene sets with both approaches shows that the parent-child approach drastically reduces the number of descendant terms falsely predicted to be over-represented.

Given this systematic analysis of the advantages of the parent-child approach we think that it should become the future standard. However, it should be clear that, since the two approaches use different statistics, the interpretation of results obtained with the term-for-term approach cannot be carried over to the parent-child approach. The following proper understanding of how the parent-child results have to be interpreted is necessary on the user's side.

One might argue that the inheritance problem of the term-for-term approach is in fact not a problem, but an advantage, since it also detects interesting descendant terms of over-represented terms which the parent-child approach would miss. Still, the parent-child approach does *not* state that those descendant terms are biologically irrelevant. It states that the experiment which resulted in the study set *does not give enough information* to claim that some of those descendant terms are more relevant than others and that therefore all descendant terms might be equally important in further studies. In turn, the additional emergence of descendant terms under the parent-child approach clearly indicates their increased importance. With that interpretation in mind one can claim that the parent-child approach gives more detailed insights into the GO annotation of the study set than the term-for-term approach.

The all-subset minimal p-values which we introduced in Subsection 3.1 are another key quantity which we think is of great importance in the context of the parent-child approach. Knowing about the parent-child combinations for which the all-subset minimal p-values are rather large gives important insights into the nature of the GO annotations of the underlying population set. We therefore plan to incorporate the all-subset minimal p-values into a visualization of the results obtained from the parent-child approach as it is produced by the Ontologizer.

We explicitly did not focus on the problem of multiple testing corrections (MTCs) in the context of finding over-represented GO terms in gene sets. Although some of the standard approaches have meanwhile been implemented and tested in this context, we think that there is still room for improvement and we will broaden our research to that topic. The problem of finding the optimal MTC is hard, because of the complicated dependencies between the GO terms which are caused by the DAG structure and by the annotation of individual genes to multiple terms. The parent-child approach corrects for some of those dependencies, but there remain other non-trivial dependencies between parent-child p-values. The parent-child approach therefore adds a new facet to the topic of MTCs, because it is not clear that the same strategy will turn out to be optimal for both approaches.

References

1. Ashburner, M., Ball, C., Blake, J., Botstein, D., Butler, H., Cherry, J., Davis, A., Dolinski, K., Dwight, S., Eppig, J., Harris, M., Hill, D., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J., Richardson, J., Ringwald, M., Rubin, G., Sherlock, G.: Gene Ontology: tool for the unification of biology. The Gene Ontology Consortium. Nature Genetics **25** (2000) 25–29

2. Harris, M.A., Clark, J., Ireland, A., Lomax, J., Ashburner, M., Foulger, R., Eilbeck, K., Lewis, S., Marshall, B., Mungall, C., Richter, J., Rubin, G.M., Blake, J.A., Bult, C., Dolan, M., Drabkin, H., Eppig, J.T., Hill, D.P., Ni, L., Ringwald, M., Balakrishnan, R., Cherry, J.M., Christie, K.R., Costanzo, M.C., Dwight, S.S., Engel, S., Fisk, D.G., Hirschman, J.E., Hong, E.L., Nash, R.S., Sethuraman, A., Theesfeld, C.L., Botstein, D., Dolinski, K., Feierbach, B., Berardini, T., Mundodi, S., Rhee, S.Y., Apweiler, R., Barrell, D., Camon, E., Dimmer, E., Lee, V., Chisholm, R., Gaudet, P., Kibbe, W., Kishore, R., Schwarz, E.M., Sternberg, P., Gwinn, M., Hannick, L., Wortman, J., Berriman, M., Wood, V., de la Cruz, N., Tonellato, P., Jaiswal, P., Seigfried, T., White, R.: The Gene Ontology (GO) database and informatics resource. Nucleic Acids Res **32** (2004) D258–D261
3. Camon, E., Magrane, M., Barrell, D., Lee, V., Dimmer, E., Maslen, J., Binns, D., Harte, N., Lopez, R., Apweiler, R.: The Gene Ontology Annotation (GOA) Database: sharing knowledge in Uniprot with Gene Ontology. Nucleic Acids Res **32** (2004) D262–D266
4. Castillo-Davis, C.I., Hartl, D.L.: GeneMerge–post-genomic analysis, data mining, and hypothesis testing. Bioinformatics **19** (2003) 891–892
5. Berriz, G.F., King, O.D., Bryant, B., Sander, C., Roth, F.P.: Characterizing gene sets with FuncAssociate. Bioinformatics **19** (2003) 2502–2504
6. Draghici, S., Khatri, P., Bhavsar, P., Shah, A., Krawetz, S.A., Tainsky, M.A.: Onto-Tools, the toolkit of the modern biologist: Onto-Express, Onto-Compare, Onto-Design and Onto-Translate. Nucleic Acids Res **31** (2003) 3775–3781
7. Beissbarth, T., Speed, T.P.: GOstat: find statistically overrepresented Gene Ontologies within a group of genes. Bioinformatics **20** (2004) 1464–1465
8. Martin, D., Brun, C., Remy, E., Mouren, P., Thieffry, D., Jacq, B.: GOToolBox: functional analysis of gene datasets based on Gene Ontology. Genome Biol **5** (2004) R101
9. Robinson, P.N., Wollstein, A., Böhme, U., Beattie, B.: Ontologizing gene-expression microarray data: characterizing clusters with Gene Ontology. Bioinformatics **20** (2004) 979–981
10. Khatri, P., Draghici, S.: Ontological analysis of gene expression data: current tools, limitations, and open problems. Bioinformatics **21** (2005) 3587–3595
11. Sharan, R., Suthram, S., Kelley, R.M., Kuhn, T., McCuine, S., Uetz, P., Sittler, T., Karp, R.M., Ideker, T.: Conserved patterns of protein interaction in multiple species. Proc Natl Acad Sci U S A **102** (2005) 1974–1979
12. Westfall, P.H., Young, S.S.: Resampling-Based Multiple Testing: Examples and Methods for p-Value Adjustment. Wiley-Interscience (1993)
13. Ge, Y., Dudoit, S., Speed, T.: Resampling-based multiple testing for microarray data analysis. TEST **12** (2003) 1–77
14. Dwight, S.S., Harris, M.A., Dolinski, K., Ball, C.A., Binkley, G., Christie, K.R., Fisk, D.G., Issel-Tarver, L., Schroeder, M., Sherlock, G., Sethuraman, A., Weng, S., Botstein, D., Cherry, J.M.: Saccharomyces Genome Database (SGD) provides secondary gene annotation using the Gene Ontology (GO). Nucleic Acids Res **30** (2002) 69–72
15. Spellman, P., Sherlock, G., Zhang, M., Iyer, V., Anders, K., Eisen, M., Brown, P., Botstein, D., Futcher, B.: Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. Mol Biol Cell **9** (1998) 3273–3297
16. Gansner, E.R., North, S.C.: An open graph visualization system and its applications to software engineering. Software — Practice and Experience **30** (2000) 1203–1233