BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 6 2013, pages 733–741
doi:10.1093/bioinformatics/btt045

Genetics and population analysis

Advance Access publication January 29, 2013

BioGranat-IG: a network analysis tool to suggest mechanisms of genetic heterogeneity from exome-sequencing data

Nick Dand[1], Frauke Sprengel[2], Volker Ahlers[2] and Thomas Schlitt[1,3,*]

[1]Department of Medical and Molecular Genetics, King's College London, London SE1 9RT, UK, [2]Faculty IV, Department of Computer Science, University of Applied Sciences and Arts Hannover, 30531 Hannover, Germany and [3]Institute for Mathematical and Molecular Biomedicine, King's College London, London SE1 1UL, UK

Associate Editor: Jeffrey Barrett

ABSTRACT

Motivation: Recent exome-sequencing studies have successfully identified disease-causing sequence variants for several rare monogenic diseases by examining variants common to a group of patients. However, the current data analysis strategies are only insufficiently able to deal with confounding factors such as genetic heterogeneity, incomplete penetrance, individuals lacking data and involvement of several genes.

Results: We introduce BioGranat-IG, an analysis strategy that incorporates the information contained in biological networks to the analysis of exome-sequencing data. To identify genes that may have a disease-causing role, we label all nodes of the network according to the individuals that are carrying a sequence variant and subsequently identify small subnetworks linked to all or most individuals. Using simulated exome-sequencing data, we demonstrate that BioGranat-IG is able to recover the genes responsible for two diseases known to be caused by variants in an underlying complex. We also examine the performance of BioGranat-IG under various conditions likely to be faced by the user, and show that its network-based approach is more powerful than a set-cover-based approach.

Availability: We implemented our methods in Java as BioGranat-IG, a bundle within our BioGranat graph analysis and visualization tool (www.biogranat.org).

Contact: thomas.schlitt@genetics.kcl.ac.uk

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on November 9, 2012; revised on January 15, 2013; accepted on January 22, 2013

1 INTRODUCTION

Improvements in sequencing technology have made sequencing the entire human exome both practical and affordable. This has led to considerable successes in the identification of disease-causing sequence variants for several rare Mendelian diseases such as Kabuki Syndrome (Ng *et al.*, 2010a) and Hajdu-Cheney Syndrome (Simpson *et al.*, 2011). Typically, whole-exome sequencing is undertaken for a small number of patients. Comparison with a reference genome yields, for each individual, a list of genes carrying sequence variants. These lists are filtered, by removing, for example, genes in which all mutations are predicted to be without severe impact on protein function, in which mutations occur too frequently relative to the prevalence of the disease or in which no mutations were observed when sequencing affected relatives (Ng *et al.*, 2010b). Finally, the filtered lists can be inspected to find their intersection. In the best cases, only one remaining gene is shared between all affected individuals, becoming a natural candidate for experimental follow-up studies, for examples see (Ng *et al.*, 2009; Simpson *et al.*, 2011) and the review paper (Ku *et al.*, 2011).

However, depending on the disease, one might not observe any genes in the intersection of the filtered lists. One phenomenon that can cause this is *genetic heterogeneity*, where one phenotypic outcome results from any one of a number of possible mutations, possibly in different loci (McClellan and King, 2010). An example would be two genes that are functionally related through their protein products, both playing a critical role in some cellular function and such that a mutation in either gene leads to the failure of that function.

One possible way to deal with this problem would be to look for the smallest set of genes such that all individuals carry a sequence variant in the set. This frames the problem as one of minimal set cover (Cormen, 2001) and will not guarantee any biological meaning or functional relatedness for the set of genes found.

Instead, in this article, we present BioGranat-IG ('BioGranat Individuals-Grouping'), a tool that uses the additional structure found in biological networks to analyse sequence data for multiple individuals and suggest possible sources of genetic heterogeneity.

Biological networks connect genes or proteins for which a functional relationship is known—or predicted—to exist. These networks can be represented as graphs. For example, an edge in a protein–protein interaction network graph connects two nodes (genes) where the proteins coded by those genes interact. Other possible graphs might connect genes based on co-expression, genomic location or cell signalling (Barabasi and Oltvai, 2004), text mining of published literature (Jenssen *et al.*, 2001) or integrated evidence for interaction based on several of the above measures (Lee *et al.*, 2011; Szklarczyk *et al.*, 2011). Several tools, such as GeneMANIA (Warde-Farley *et al.*, 2010) and STRING (Szklarczyk *et al.*, 2011), exist that use biological networks to suggest additional functionally related genes for a given input gene list. Conversely, our tool will use the networks to focus on which of the input genes are likely to be causative and provide a relatively short list of candidate genes for follow-up study.

*To whom correspondence should be addressed.

The premise behind BioGranat-IG is that if a disease has an underlying mechanism of genetic heterogeneity, the genes involved may be functionally related and therefore closely connected in a biological network; if not directly connected, they might be interacting with one or more common neighbours. To identify disease gene candidates, given a number of individuals and for each individual a list of genes with sequence variants (filtered as described above), we mark each node in our network with the individuals who carry a mutant version of that gene. Subsequently, we seek the smallest connected subnetwork marked with all individuals.

Our network thus consists of 'marked' nodes, representing genes where a sequence variant was observed in some individual, and 'empty' nodes, where no mutation was observed for any individual. As an additional constraint on the subnetwork we seek, we require that each marked node is connected to another marked node via at most one empty node. That is, the subnetwork is allowed to incorporate 'jumps' of one empty node, but not more (see Fig. 1). The rationale underlying this is that biological networks tend to exhibit the small world phenomenon, meaning that the average path length between any two nodes is short, typically four to six edges (Xu *et al.*, 2011). We are therefore interested only in localized connections, and allowing too many empty nodes into our subnetwork could reduce the chance of it having any biological meaning. This constraint allows a computationally efficient implementation, making it possible to run BioGranat-IG on large sets of permuted data to establish statistical significance.

We will present the methods implemented by BioGranat-IG and demonstrate the validity of our approach using simulated exome-sequence output. We then analyse the performance of the tool under various conditions, before considering the outstanding challenges of this approach.

To our knowledge, BioGranat-IG is the first tool developed to tackle the problem of finding disease-causing genes from exome-sequence data for Mendelian diseases with heterogenetic background. While finding dysregulated subgraphs is an intensively studied problem (Lehne and Schlitt, 2012; Staiger *et al.*, 2012), few approaches consider data for individuals separately. Rather, most approaches work with summary statistics. Notably, approaches by Dao *et al.* (Dao *et al.*, 2010), DEGAS (Ulitsky *et al.*, 2010) and KeyPathwayMiner (Alcaraz *et al.*, 2012) use differential expression data for individuals to find subnetworks containing genes differentially expressed in patients versus controls. While for differential gene-expression data one usually expects to find clusters of co-expressed functionally related genes, our problem differs because we expect all individuals to carry only a limited number (probably less than tens) of disease causing genes hidden among a large number (hundreds) of variants not related to the disease of interest. Therefore, the problem addressed by KeyPathwayMiner and DEGAS is similar to the problem we address here in general, but there are important differences in the detail that have an impact on the algorithm design.

The DAPPLE tool (Rossin *et al.*, 2011) prioritizes genes in genomic regions associated with a disease using a network-based approach conceptually related to BioGranat-IG. However, DAPPLE is designed to improve understanding of disease-associated loci, and is not readily applicable to the sequencing problem we describe.

Fig. 1. Subnetworks returned by BioGranat-IG can connect marked nodes via jumps of one empty node (a gene in which no individuals have a mutation) but not more. These examples use individuals labelled A–D

2 METHODS

BioGranat is a software tool for the analysis and visualization of biological networks, developed in collaboration by Hochschule Hannover and King's College London (Mendig *et al.*, 2009). It is implemented using the Java OSGi framework and is freely available from www.biogranat.org. BioGranat-IG has been developed as a BioGranat bundle.

In graph-theoretic terms, the problem we face can be expressed as follows. Let S_n be the set of n elements $\{1,2,\ldots,n\}$. The power set $P(S_n)$ of S_n is the set of all possible subsets of S_n, including the empty set. Then, given a graph G with vertices $V(G)$, and for each v in $V(G)$ a mapping $f(v)$ into $P(S_n)$, we wish to find the smallest connected subgraph G' of G such that

$$\bigcup_{v \in V(G')} f(v) = S_n$$

Here, smallest is taken to mean least number of vertices. It is possible that no such subgraph exists, in which case we seek the smallest connected subgraph such that

$$\left| \bigcup_{v \in V(G')} f(v) \right| = m$$

where m is the maximum number of elements of S_n that are mapped to by the vertices of a single connected component of G. (Note that in this article, we refer more loosely to seeking small subnetworks 'containing' all individuals).

The problem is an example of the minimal connected set-cover problem (MCSC) (Cerdeira and Pinto, 2005; Zhang *et al.*, 2009), which is NP-hard because it is a generalization of the minimal set-cover problem (MSC) (Karp, 1972). Several authors have published approximation algorithms for MCSC in recent years (Elbassioni *et al.*, 2012; Ren and Zhao, 2011). However, for BioGranat-IG, we have developed a new method because we want to collect not just the size of the optimal subnetwork, but all examples of optimal and near-optimal subnetworks, up to a user-specified size. BioGranat-IG cannot determine which subnetworks will be of most interest to the user, and so must output them all.

We will not devote further space to a discussion of the complexity of the problem, but instead present the methods used by BioGranat-IG to find near-optimal small subnetworks containing mutations for maximum individuals.

2.1 Network pre-processing

BioGranat-IG works by marking lists of genes for multiple individuals onto a biological network. There are >20 000 human genes but the function and expression of many genes is only poorly understood. Therefore, currently available networks are all incomplete. Nevertheless, they contain thousands of nodes and edges. To speed computation, the network is pre-processed (see Supplementary Fig. S1).

- Because 'jumps' of more than one empty node are not allowed, all edges with an empty node at both ends are removed. From this point on, any neighbour of an empty node must be a marked node.
- All empty nodes of degree zero or one are removed.
- Where two empty nodes are connected to the same set of neighbours, one of the nodes can be removed from the network (and stored to provide an alternative result should the kept node turn out to form part of a minimal subnetwork).
- Any empty node whose neighbours form a clique (a complete subnetwork) is removed from the network. Such a node will never be called on to link two marked nodes.

2.2 Triplet and quadruplet search

Before resorting to heuristic methods, which cannot guarantee that all minimal subnetworks are returned, BioGranat-IG performs two searches (triplet search and quadruplet search), which together comprise an exhaustive search of all subnetworks of up to four nodes. If a subnetwork is found that covers all individuals, there is no need to run any subsequent searches.

The triplet search identifies all candidate subnetworks of up to three nodes, using the fact that for three nodes to be connected there must be at least one path of length two connecting them. We call the subnetwork induced by such a path a triplet. For each node in the network (whether marked or empty) we first check whether this node alone contains all individuals, and then identify the neighbouring nodes. All pairs formed from the original node plus one marked neighbour, and triplets formed from the original node plus two marked neighbours are examined. At this point, if a subnetwork is found containing all individuals, there is no need to continue with the quadruplet search.

The quadruplet search builds on the triplet search. Using the constraint that only one empty node can be 'jumped' at a time, we know that for a subnetwork of four nodes to minimally cover all individuals it must contain at least one triplet with two individuals. Using this set of triplets as a starting point, quadruplets are thus constructed through the addition of any neighbouring nodes that confer additional individuals.

2.3 Minimum-distance search

If the triplet and quadruplet searches fail to find a subnetwork covering all individuals, BioGranat-IG will perform heuristic searches based on a minimum-distance approach (described here) and a multi-minimum-distance approach (described in the next section).

The *minimum-distance search* uses a greedy approach to build subnetworks starting from a single node. The selection function used to determine the most valuable neighbour to add to the subnetwork at each step is the sum of the minimum distances (length of shortest path) from each neighbour to all individuals not already covered by the subnetwork.

This approach requires that for every node in the network, the minimum distance to each individual is calculated. For node v, the distances $\{d_1(v),\ldots,d_n(v)\}$ represent the minimum distance from v to any node that contains individual $1,\ldots,n$, respectively. Distances are calculated using a multi-source breadth-first search approach, as described by the following pseudocode:

```
1: For each individual, i
2:    For each node v in the component
3:       If v is marked with i, set d_i(v) = 0, and add v to the queue
4:       Else set d_i(v) = ∞
5:    While the queue is not empty
6:       Take node v from the queue, and for all neighbours v' of v
7:          If d_i(v') = ∞, set d_i(v') = d_i(v) + 1, and add v' to the queue
```

Because the pre-processed network may consist of several components, infinite distances can remain.

The search proceeds, only in components that contain sufficiently many individuals, by recursively building up subnetworks starting from a single node. The basis for recursion is as follows: in a component containing individuals $I=\{i_1,\ldots,i_{n'}\}$, then given a subnetwork G' of that component containing individuals $J=\{j_1,\ldots,j_m\}$ $(m<n')$, examine all neighbours of nodes in G'. Of these neighbours, find the node v that minimizes the following sum:

$$\sum_{i\in I\setminus J} d_i(v)$$

Form a new subnetwork, G'' by adding v to G', and repeat until all n' individuals are incorporated.

At each step, there may be a tie amongst neighbours for the smallest minimum distance sum, and in this case each alternative G'' is explored in turn. In practice this occurs frequently, leading to many calls of the recursive function in what is effectively a depth-first search strategy (Cormen, 2001).

The results of this search depend on the starting node chosen, so the approach taken is to use all nodes in a component as starting nodes. This is not as costly as it may first appear owing to several steps that are taken to ensure the search runs efficiently:

- To avoid duplication of effort, a list of all subnetworks explored is kept. Suppose a search starting at node v adds node w first. If then the search starting at node w were to add node v first, the search will stop because the subnetwork v-w has already been explored.
- The size, s, of the smallest subnetwork found containing all reachable individuals is maintained. Subsequently, node v will not be added to subnetwork G' to form G'' unless v is within distance $s-|G''|$ of one of the individuals needed by G'.
- Nodes are used as starting nodes in order (from smallest to largest) of their total minimum distance to all reachable individuals so that the smallest subnetwork size is found as quickly as possible.
- The number of calls of the recursive function is limited to 1000 from each starting node, preventing excessive worst-case running times due to many ties between neighbours. Hitting 1000 calls would make it highly unlikely that our starting node is part of a small subnetwork of interest to us.

2.4 Multi-minimum-distance search

Although the minimum-distance search often works well, it does not guarantee finding the optimal subnetwork. One reason for this is that no credit is given for the fact that a neighbour might extend a subnetwork towards more than one individual. If a node is labelled with individuals 1 and 2, then its neighbour v has $d_1(v)+d_2(v)=2$, overstating the actual distance. Figure 2 gives an example where the minimum distance search would not find the optimal subnetwork, regardless of which node is used as a starting node.

The multi-minimum-distance search partially addresses this problem by preferentially seeking nodes with multiple individuals during the recursive step. The search runs recursively in the same way as the minimum-distance search, with the only difference being the definition of the distances used in the sum when selecting the best neighbour of a subnetwork.

Having previously defined the simple minimum distance $d_i(v)$, we now introduce the multi-minimum distance $d_{i,k}(v)$ for $k=1,\ldots,n$. This is defined as the length of the shortest path from v to any node that is marked with $\geq k$ individuals, such that one of those individuals is individual i. If no such marked node exists (that is, the component under consideration does contain individual i, but not in any node with $\geq k$ individuals), then we set $d_{i,k}(v)=d_{i,k-1}(v)$. This is always well-defined

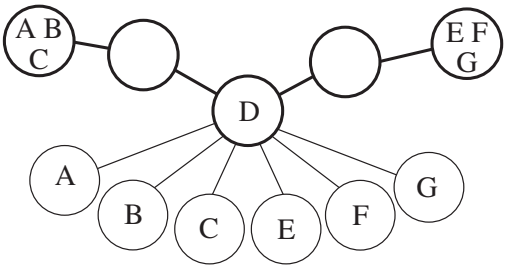

Fig. 2. An example of a network for which the minimum distance search would fail to find the smallest subnetwork containing all individuals A–G. Nodes are labelled with the individuals attached to them. The true optimal subnetwork comprises the uppermost five nodes, indicated by thicker lines. However, nodes from the bottom row will be incorporated into any subnetwork found by the minimum distance search, regardless of which node the search starts from. Note that in this case the multi-minimum-distance search *would* find the optimal subnetwork

because $d_{i,1}(v)$ is equivalent to the simple minimum distance $d_i(v)$. Distances are calculated in the same way as the simple minimum distance, for one value of k at a time, starting with $k=1$.

The search proceeds recursively as before, only now given a subnetwork G' containing individuals $J=\{j_1, \ldots, j_m\}$ $(m<n')$, the next node v is the neighbour that minimizes the following sum:

$$\sum_{i \in I \setminus J} d_{i,n'-m}(v)$$

The reasoning behind this approach is that when few individuals have been found, it is beneficial to extend the subnetwork towards nodes with multiple individuals. Conversely, later in the search, if only one more individual is sought, the nearest node that contains it will do.

Although this search recognizes that nodes with multiple individuals are important, it is not always efficient. For example, suppose individual i is one of three individuals still needed by a subnetwork. The distance $d_{i,3}(v)$ does not necessarily give the distance from node v to a node containing those three individuals, but just the distance from v to a node containing *any* three individuals, one of which is individual i. To know the former distance would effectively require the distances be recalculated at each recursion. This is prohibitively expensive, yet provides no guarantee of finding the optimal subnetwork.

If there is a small subnetwork in which two or three nodes contain most of the individuals (which is feasible biologically), the multi-minimum-distance search is likely to find it, albeit a simple extension of the minimum-distance search.

2.5 Program output and user options

Sometimes several minimal subnetworks are found that overlap (have nodes in common). Suppose subnetwork v_1-v_2-v_3 is the true underlying cause of a disease, and all individuals have a mutation in one of these genes. It could be the case, by chance, that some of the individuals also have mutations in a connected gene v_4, such that v_1-v_2-v_4 also covers all individuals. Equally, it could also be true that elsewhere in the network, three different connected genes v_5-v_6-v_7 cover all individuals by chance. BioGranat-IG can group overlapping subnetworks and return the resulting 'groups' of nodes. In this case, two groups would be returned, v_1-v_2-v_3-v_4 and v_5-v_6-v_7 (along with the frequency of inclusion for each node, to quantify its importance in the group; the first group here would have a count of 2 for v_1 and v_2, and 1 for v_3 and v_4). Thus, we provide candidate genes for experimental follow-up studies to determine the true disease-causing genes.

In BioGranat-IG, the criteria for 'optimal' subnetworks can be relaxed by tolerating more nodes, or fewer individuals, up to user-specified limits. This flexibility allows the user a fuller analysis of potentially interesting results. In addition, there is a parameter for maximum subnetwork size, which will limit the size of any subnetworks found. This is useful in the situation where the smallest subnetwork containing all individuals is large: we might be interested in whether there exist much smaller subnetworks that contain most (rather than all) of the individuals.

In the typical situation where a small subnetwork is found that covers all individuals, BioGranat-IG offers the functionality to test whether this subnetwork is a significant finding. This can be done by generating random gene lists having the same number of genes in the network as the original gene lists. For each random instance the searches are run, and the significance of the original subnetwork found can be measured by the frequency with which equally small or smaller subnetworks cover all individuals in the random simulations.

It is worth mentioning at this stage that we can still construct examples of networks labelled with individuals for which none of the methods described would find the optimal subnetwork (for example, see Supplementary Fig. S2). But these counter-examples are much larger and more contrived, and it would seem unlikely that such a region would be biologically relevant.

3 RESULTS

In this section, we firstly demonstrate that the principle underlying BioGranat-IG is sound and that the program produces valid results, using two diseases known to have a basis of genetic heterogeneity. We show that BioGranat-IG can recover the genes responsible for Acne Inversa (AI) and Pseudohypoaldosteronism type I (PHA-I) using simulated exome-sequencing data. We then examine the performance of BioGranat-IG under various conditions, including the nature of the underlying disease complex and the amount and quality of input data.

All testing has been performed on a human protein–protein interaction network (huppi2) derived as described earlier (Lehne and Schlitt, 2009) by integration of data from six public databases. It is undirected, self-loops have been removed and each interaction has been reported in two or more scientific publications. See Table 1 for network properties. The choice of network to use with BioGranat-IG, as with other network-based analysis methods, involves consideration of competing factors. There is typically a trade-off between network coverage (number of genes represented in the network) and the degree of confidence that can be placed on network interactions. The choice of network will also be influenced by whether a particular type of genetic mechanism is predicted and by the sequence data available. When sequencing small groups of affected individuals, it makes sense to run BioGranat-IG on smaller high-quality networks initially (this minimizes the risk of connecting the individuals using false-positive connections, as would be more likely in a larger network) and proceed to larger networks if no positive results are found. Smaller networks also have the added advantage of reduced computation time. The huppi2 network is a high-confidence network, but note that the tests on simulated AI and PHA-I data were repeated in a range of other networks of varying size with comparable results. See the Supplementary Material for additional results.

3.1 Testing methodology and metrics

All tests examine how well BioGranat-IG can recover a specified gene complex in 1000 simulations. For each simulation, we

Table 1. Properties of the huppi2 network

Property	Value
Number of nodes (genes)	3666
Number of edges (interactions)	6187
Number of components	244
Average degree	3.38
Maximum degree	108
Average pairwise shortest-path length	5.88
Maximum pairwise shortest-path length	23

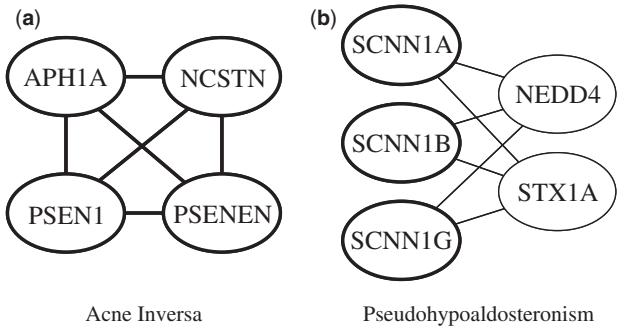

Fig. 3. The subnetworks in huppi2 that contain the genes responsible for the two positive control diseases. (**a**) AI has four underlying genes that form a clique. (**b**) Pseudohypoaldosteronism has three underlying genes (thick lines) that are not connected to each other but can be connected via one of two connecting genes (thin lines)

randomly generate lists of variant-containing genes for a fixed number of individuals. Unless otherwise stated, we use 15 individuals per simulation, which is a typical sample size. Each simulated individual is generated by randomly picking one gene from within the complex of interest and a fixed number of non-causal genes from the rest of the huppi2 network. Unless otherwise stated, we generate 35 non-causal nodes per individual. This number corresponds to the typical number of candidate genes per individual generated by exome sequencing (after filtering) (Ng *et al.*, 2009; Simpson *et al.*, 2011) that map to huppi2. We refer to this process as generating *random individuals* and *spiking in* the complex of interest.

For illustrative purposes, suppose we choose to spike in a complex of five genes. Random selection with replacement gives no guarantee that all five will be spiked into a given set of 15 individuals (in fact, the probability is only 0.83). We do not force all five to be spiked in, as there would be no such guarantee in practice with real patients' data. Therefore, one of the metrics we look at in each test is the *number of nodes actually spiked in*.

The key result is the *number of spiked nodes recovered*, which is the number of genes from the complex of interest that are returned in the output of BioGranat-IG. This can exceed the number actually spiked in. For example, if only four of a complex of five nodes are actually spiked in, it is possible that the fifth 'true' node could be found as a 'jump'.

In addition, we consider the *number of false nodes returned*. These are nodes that do not form part of the complex of interest, but are nevertheless returned in the output of BioGranat-IG. This can occur when, by chance, nodes neighbouring the complex are marked with individuals in such a way that an alternative smallest subnetwork can be formed by excluding one of the 'true' genes in the complex, and including the non-causal neighbour. Because BioGranat-IG has no prior knowledge of the 'true' disease-linked genes, all genes found are returned.

All numbers referred to in the results section are the average values found in 1000 simulations.

3.2 BioGranat-IG recovers acne inversa genes

AI, an inflammatory skin disease, has been shown to result from a mutation in the γ-secretase complex comprising the genes APH1A, NCSTN, PSEN1 and PSENEN (see www.omim.org; MIM ID #142690). Mutations in three of these genes have been directly linked to AI (Wang *et al.*, 2010).

Using BioGranat-IG on this complex, we were able to recover all four genes in 957 of the 1000 simulations, but a more detailed examination of the results gives more insight into the performance.

Of the 1000 simulations, 43 had three or fewer of the four AI genes: on average the number of nodes actually spiked in was 3.957. Of these 43 simulations, none resulted in the recovery of all four AI genes. However, for every simulation, every gene that was actually spiked in was recovered, so the number of spiked nodes recovered was also 3.957. Whether any unspiked 'true' nodes are recovered depends on the topology of the network around the spiked nodes. In the case of AI, the four genes form a clique in huppi2, and consequently, any three of the four form a connected subnetwork (see Fig. 3a). So if only three of the genes are marked with individuals, there is no need for the fourth to be incorporated into the optimal subnetwork as a 'jump'.

The average number of false nodes found was 0.021 (one false node in 21 of the 1000 simulations). For AI, then, BioGranat-IG would be highly likely to direct the user towards the true causal complex, with minimal erroneous results.

3.3 BioGranat-IG recovers PHA-I genes, plus 'jumps'

The second disease used as a positive control is PHA-I, a disorder of electrolyte metabolism affecting infants. Studies have shown that affected individuals can exhibit mutations in any of the genes SCNN1A, SCNN1B and SCNN1G (see www.omim.org; MIM ID #264350).

In huppi2, these genes are not directly connected; although there is evidence for direct protein interactions (Firsov *et al.*, 1996), it does not meet the quality criteria for inclusion in huppi2. However, all three genes are connected to NEDD4 and to STX1A (Fig. 3b).

The average number of nodes actually spiked in was 2.990: all three genes were spiked in for 990 of the 1000 simulations. In 989 of these, all three were correctly recovered by BioGranat-IG.

In one case, all three PHA-I genes were spiked in but only two were recovered. This occurred because 2 of the 15 individuals were spiked with gene SCNN1G, but for both of these individuals, STX1A happened to be one of the 35 non-causal variants generated. As they are not directly connected, any subnetwork including all three PHA-I genes requires at least four nodes, but

in this case, a smaller subnetwork can be formed from SCNN1A, SCNN1B and STX1A, so this smaller subnetwork is returned. Owing to this one simulation run, the average number of spiked nodes recovered was lower than the average number of nodes actually spiked, at 2.989.

Because an additional node is always needed to connect the PHA-I genes, and there are two alternative ways to do this, both NEDD4 and STX1A are always returned with the PHA-I genes found (other than the anomalous case described). Therefore, the average number of false nodes found is relatively high at 2.009 (this includes 10 simulations where an additional false gene was returned). However, this is a positive result. It shows that BioGranat-IG can work successfully even when the network used does not contain the true causal genes as a connected complex. Note that returning a small number of false genes and/or 'jumps' is not hugely problematic, as BioGranat-IG is intended to be used to highlight genes for further experimental investigation.

3.4 The effectiveness of BioGranat-IG depends on a number of conditions

Having used real diseases to show that BioGranat-IG can find sources of genetic heterogeneity, we now examine the performance of BioGranat-IG under various conditions using artificial data.

3.4.1 Smaller, less connected complexes give better results The ability of BioGranat-IG to find a spiked-in complex and the number of false genes it is likely to return depend on both the size of the complex and the local network topology.

To measure this, we identified three complexes of seven nodes each in the huppi2 network. Complex L-7 has low connectivity (it forms a 'Y'-shaped 'branch' with a single neighbour in the rest of the network); complex A-7 has average connectivity (each node has degree 3 or 4) and complex H-7 has high connectivity (each node has degree >25). Simulations were run using subcomplexes of between two and seven nodes from each of these complexes (for example, A-5 being the subcomplex of A-7 that has five nodes).

Results for the low-connectivity complexes were excellent, with frequent recovery of additional nodes from the complex not actually spiked in, and few false nodes returned (see Table 2a and Fig. 4a). For L-7, on average, only 6.315 nodes from the complex are actually spiked in, but 6.574 are recovered. In addition, only 0.746 false nodes are returned. The performance improves as the size of the complex gets smaller. This is probably owing to the reduced chance of alternative smallest subnetworks forming elsewhere by chance and reduced chance of a 'true' node not actually being spiked in.

For the average-connectivity complexes, the performance suffers a little because having more nodes neighbouring the complex gives more opportunities for false variants to occur by chance in close proximity to the 'true' disease nodes, thus offering alternative ways to form small subnetworks. However, the number of nodes recovered for the A complexes is broadly in line with the L complexes, while the numbers of false nodes returned are only slightly higher and still tolerable in practice (see Table 2b and Fig. 4b).

The results for the high-connectivity complexes show a more substantial impact due to the presence of so many more neighbours, all of which can potentially be included as false variants (see Table 2c and Fig. 4c). The power to recover genes in the H complexes is slightly reduced relative to the A complexes (for H-7, 6.322 nodes are actually spiked in but only 6.297 recovered). This reflects the increased 'noise' around the complex making the true signal harder to detect. However, the bigger impact is to the number of false nodes returned, which is, for example, 4.665 for H-6 and 4.657 for H-7 (intuitively this would increase with the size of the complex—it most likely does not owing to the particular topologies of H-6 and H-7).

In summary, when the underlying complex happens to be in a highly connected part of the network, the total number of nodes found is generally higher. Because the goal is to find an unknown complex, its connectivity is not something that will be known to the experimenter *a priori*, but fortunately sequencing more individuals can help combat this problem. We tested H-6 and H-7 again, this time using 30 individuals, and the results were much improved (less than one false node found on average in each case).

3.4.2 Sequencing more individuals can improve results To fully characterize the relationship between performance and number of individuals, we ran simulations with varying number of individuals on complex A-5 (to represent a typical complex). The results, given in Table 2d and Figure 4d, confirm that a higher number of individuals leads to increased power to detect the spiked complex and fewer false nodes being returned. There are two reasons for this: the increased chance of having all nodes in the complex actually spiked in, and the reduced chance of sufficient false nodes occurring in the nodes neighbouring the region to offer alternative small subnetworks.

Clearly, the conclusion that can be drawn here is that increasing the sample size increases the power to detect a true disease-linked signal. This suggests a strategy that could be followed when BioGranat-IG is used in practice: if the number of genes returned is large, sequencing further individuals will help to narrow this list down if there is a true underlying cause of genetic heterogeneity.

3.4.3 Stringency of variant filtering affects performance As previously described, it is common practice when searching for genetic causes of rare diseases using exome-sequencing data to filter out variants found in patients based on a number of criteria (e.g. genes that are well understood or in which variants are seen frequently). More stringent filtering should reduce the number of false input nodes per individual, which in turn affects the performance of BioGranat-IG.

We tested this by running simulations on complex A-5 with varying number of false nodes per individual, but for this complex, we found a marginal effect (results not shown).

The effect can be seen more clearly in the equivalent simulations run on complex H-7 (see Table 2e, Fig. 4e). As expected, for 35 false variants, the results are close to what we saw for the same complex in section 3.2.1. With a change in the number of false input nodes per individual, there is again a marginal effect on the number of spiked nodes recovered, ranging from 6.332 with 15 false variants per individual down to 6.183 with 55 false

Table 2. Performance testing for BioGranat-IG

	Complex size	Actually spiked	Recovered	False positives	Total nodes
(a) Test performance on complexes of varying size in low-connectivity region (15 individuals, 35 false nodes per individual)					
Complex L-2	2	2.000	2.000	0.000	2.000
Complex L-3	3	2.994	2.996	0.000	2.996
Complex L-4	4	3.953	3.981	0.000	3.981
Complex L-5	5	4.830	4.888	0.021	4.909
Complex L-6	6	5.590	5.741	0.081	5.822
Complex L-7	7	6.315	6.574	0.746	7.320
(b) Test performance on complexes of varying size in average-connectivity region (15 individuals, 35 false nodes per individual)					
Complex A-2	2	2.000	2.000	0.000	2.000
Complex A-3	3	2.996	2.998	0.000	2.998
Complex A-4	4	3.948	3.961	0.015	3.976
Complex A-5	5	4.826	4.881	0.025	4.906
Complex A-6	6	5.604	5.789	0.111	5.900
Complex A-7	7	6.277	6.568	0.761	7.329
30 individuals					
Complex A-6	6	5.973	5.988	0.003	5.991
Complex A-7	7	6.911	6.964	0.017	6.981
(c) Test performance on complexes of varying size in high-connectivity region (15 individuals, 35 false nodes per individual)					
Complex H-2	2	2.000	2.000	0.000	2.000
Complex H-3	3	2.991	2.993	0.043	3.036
Complex H-4	4	3.950	3.962	0.395	4.357
Complex H-5	5	4.823	4.847	1.264	6.111
Complex H-6	6	5.588	5.660	4.665	10.325
Complex H-7	7	6.322	6.297	4.657	10.954
30 individuals					
Complex H-6	6	5.977	5.984	0.401	6.385
Complex H-7	7	6.938	6.938	0.943	7.881
(d) Test performance with varying number of individuals (complex A-5, 35 false nodes per individual)					
5 individuals	5	3.388	3.611	1.682	5.293
10 individuals	5	4.466	4.669	0.163	4.832
15 individuals	5	4.842	4.900	0.028	4.928
20 individuals	5	4.931	4.961	0.007	4.968
25 individuals	5	4.985	4.991	0.006	4.997
30 individuals	5	4.994	4.995	0.000	4.995
(e) Test performance with varying stringency of filtering (number of false nodes per individual) (complex H-7, 15 individuals)					
15 false nodes	7	6.307	6.332	2.068	8.400
25 false nodes	7	6.332	6.325	3.359	9.684
35 false nodes	7	6.300	6.274	4.644	10.918
45 false nodes	7	6.328	6.287	6.395	12.682
55 false nodes	7	6.270	6.183	7.761	13.944
(f) Test performance when each individual is not guaranteed a mutation in the complex, but has one with a fixed probability (complex A-5, 15 individuals, 35 false nodes per individual, no limit on sub-network size)					
Probability 0.5	5	3.977	2.241	27.565	29.806
Probability 0.6	5	4.257	3.227	22.211	25.438
Probability 0.7	5	4.474	3.953	18.241	22.194
Probability 0.8	5	4.614	4.392	13.618	18.010
Probability 0.9	5	4.742	4.718	7.352	12.070
Probability 1.0	5	4.806	4.878	0.033	4.911
Sub-networks limited to size 10					
Probability 0.5	5	3.944	1.931	13.946	15.877
Probability 0.6	5	4.251	2.951	12.890	15.841
Probability 0.7	5	4.463	3.769	10.731	14.500
Probability 0.8	5	4.622	4.421	9.248	13.669
Probability 0.9	5	4.747	4.723	5.896	10.619
Probability 1.0	5	4.832	4.895	0.054	4.949

Note: All numbers shown represent average of 1000 simulations. *Complex size* = size of complex chosen to be spiked in (number of nodes). *Actually spiked* = number of nodes in the complex picked for a given simulation (for each individual, one node in the complex is picked randomly with replacement). *Recovered* = number of nodes in the complex returned in the program output. *False positives* = number of nodes outside the complex returned in the program output. *Total nodes* = total number of nodes returned in the program output (= *recovered* + *false positives*).

variants. But the change in the number of false nodes returned is more dramatic: at 15 false variants per individual, only 2.068 are returned on average; this rises to 7.761 at 55 false variants per individual.

This confirms the intuitive notion that if there is a true disease-linked complex, filtering out more false nodes from the input gene lists will result in fewer false-positive results in the BioGranat-IG output. Of course, the user should also be aware of the risk of erroneously filtering out variants that form part of the true complex.

3.4.4 There is less chance of finding the 'true' complex if some individuals lack a variant It is of course possible that the exome-sequence data will not contain a variant for every affected individual in what is nevertheless a true mechanism of genetic heterogeneity. For example, this could be due to alternative disease pathways not present in the network, data problems (such as incorrect base calling or incomplete exome sequencing) or some individuals being phenocopies (exhibiting a similar phenotype due to environmental effects). We simulated this on complex A-5: along with 35 false nodes, each individual received a random node from complex A-5, with probability p. We tested a range of p from 0.5 to 1.0.

Note that this also provides a good model for the situation where a true disease-linked complex is only partially represented in the underlying network. This could be the case because a true gene is not present in the network or equally because true genes are disconnected in different network components (these are problems common to many network-based analysis methods).

Initially, subnetworks of any size were allowed (see Table 2f). We found that as p decreases, it is more difficult for BioGranat-IG to pick out the true underlying complex, leading to relatively poor recovery of spiked nodes for low p. Worse still, the number of false nodes found grows quickly (e.g. to over 27 at $p = 0.5$).

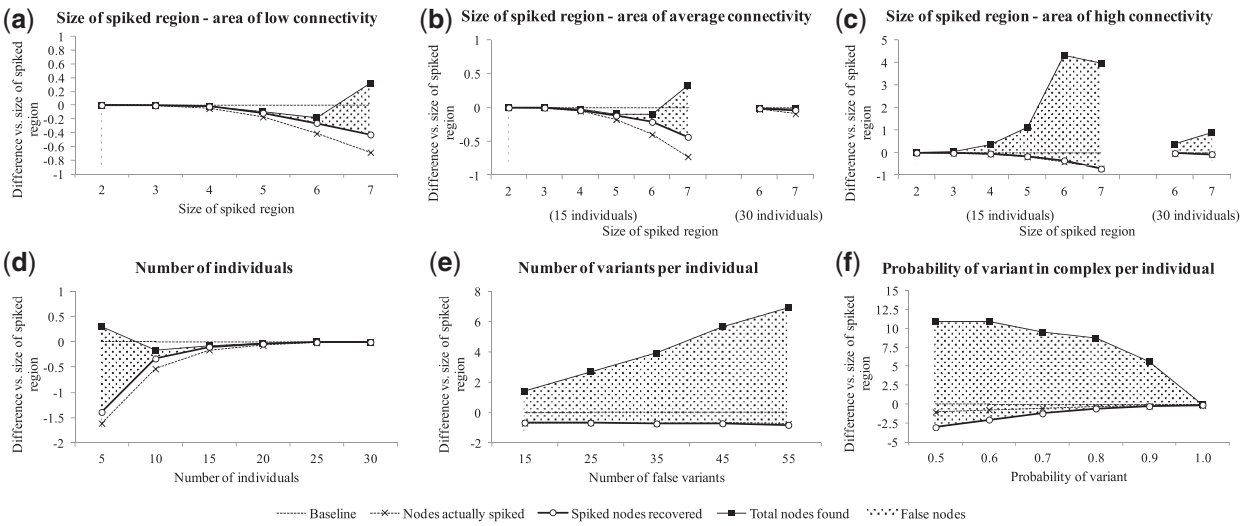

Fig. 4. Graphs depicting the performance of BioGranat-IG on simulated data in various scenarios. For each graph, the vertical axis gives the value of each metric against the size of the complex being spiked in, and metrics represent the average value observed in 1000 simulations. For example, if the spiked-in complex has four nodes and BioGranat-IG recovered 3.75 of these on average over 1000 simulations, this would be displayed at −0.25 on the vertical axis. Each graph shows a baseline at 0. Note that graphs have different scales. (**a**) Shows the ability to recover a spiked complex that falls in an area of low connectivity in the underlying network, for various complex sizes; 15 individuals, 35 false nodes per individual. (**b**) Shows the same in an area of average connectivity, for 15 individuals and 35 false nodes per individual. Also shown is the improved performance for larger complexes achieved by increasing the number of individuals to 30. (**c**) Shows the same in an area of high connectivity. (**d**) Shows the effect of using a different number of individuals. Complex is size 5, average connectivity, with 35 false nodes per individual. (**e**) Shows the effect of using a different filtering stringency (i.e. number of false nodes per individual). Complex is size 7, high connectivity, with 15 individuals. (**f**) Shows the effect of changing the probability that each individual has a variant in the complex. Complex is size 5, average connectivity, 15 individuals, 35 false variants per individual, with the maximum subnetwork size limited to 10

However, better results are obtained by limiting the maximum subnetwork size (see Table 2f and Fig. 4f). Without this limit, BioGranat-IG finds subnetworks containing all 15 individuals, no matter how big they may be. But it would be unreasonable in practice to expect that a large subnetwork identified using just 15 patients would be a true mechanism of genetic heterogeneity.

4 CONCLUDING REMARKS

We have presented BioGranat-IG, a software tool for the analysis of exome-sequencing data with the aim of identifying groups of genes in biological networks collectively responsible for causing a disease through genetic heterogeneity.

The tool addresses the problem where several patients affected by a rare Mendelian disease are exome sequenced, but no single gene is found to carry a sequence variant for all patients. It would be possible to solve the minimal set covering problem, without using a gene network, to find the smallest number of genes across which all patients have at least one variant. However, there are two advantages to using BioGranat-IG to instead perform this search within a network. Firstly, the resulting subnetwork will be made up of genes that have already been shown to interact, so is more likely to be biologically meaningful. Secondly, the number of patients needed for results to be significant is lower in the network context, where significance is measured as the likelihood of finding an equivalently small covering set of genes by chance (see Supplementary Table S1).

Using simulated datasets for two diseases, we have shown that BioGranat-IG is capable of identifying the genes known to be responsible for disease phenotype. In addition, we have shown that under a range of conditions, BioGranat-IG is generally capable of picking out a relatively small subnetwork for which further experimental investigation is likely to prove insightful. Depending on the particular disease mechanisms, it is possible to use different types of networks for the analysis. For example, to identify causal genes for metabolic diseases, a metabolic network might be more informative than a protein–protein interaction network.

Owing to the highly interconnected nature of gene networks, we have seen that false-positive genes can be suggested by BioGranat-IG, particularly when a causal gene complex is not fully contained in the network or when there might be alternative disease pathways. However, the number of genes returned will be relatively small compared with the number of variants identified by the initial exome sequencing, and can typically be inspected manually. In addition, BioGranat-IG provides a tool to estimate the significance of the results and configurable parameters to allow flexibility of the subnetworks returned, and the visualization tools in BioGranat can be used to explore the results further.

If there is a true underlying complex, sequencing more individuals should reduce the number of false-positive genes returned. It is important to note that many diseases cannot be linked to only a small number of genes, in which case BioGranat-IG may not be an appropriate tool to identify causative genes. In this case, sequencing more individuals should only increase the number of genes returned, rather than focusing in on a particular region. We do feel, however, that BioGranat-IG could prove useful for complex diseases in certain cases, for example, to

study high-severity/early-onset cases or patients having a particular subphenotype.

We are aware that in its current form, BioGranat-IG represents only a first step towards solving this problem. In particular, the methods used are likely to find the smallest connected subnetwork in which all sequenced patients have a mutation, but they do not guarantee it—we are interested in improving the algorithms to minimize the possibility of missing an optimal subnetwork. In addition, further work is to be done to improve the performance of the tool for complexes found in highly connected regions of the network, and in cases where there is a reasonable chance of affected individuals *not* having a mutation in a true underlying complex.

ACKNOWLEDGEMENTS

We would like to thank Benjamin Lehne, Natalie Prescott, Michael Simpson, Laura Southgate and Russel Sutherland for critical discussions and for access to exome-sequencing data (M.S & L.S.). Additional thanks to Nikolaos Barkas for coding advice.

Funding: Financial support by King's College London (to N.D. and T.S.), Deutscher Akademischer Austausch Dienst (DAAD) (PPP D/07/09921 to F.S. and V.A.), the British Council (ARC 1297 to T.S.) and the Royal Society (RG100252 to T.S.) is gratefully acknowledged.

Conflict of Interest: none declared.

REFERENCES

Alcaraz,N. *et al.* (2012) Efficient key pathway mining: combining networks and OMICS data. *Integr. Biol.*, **4**, 756–764.

Barabasi,A.L. and Oltvai,Z.N. (2004) Network biology: understanding the cell's functional organization. *Nat. Rev. Genet.*, **5**, 101–113.

Cerdeira,J.O. and Pinto,L.S. (2005) Requiring connectivity in the set covering problem. *J. Comb. Optim.*, **9**, 35–47.

Cormen,T.H. (2001) *Introduction to Algorithms*. MIT Press, Cambridge, MA.

Dao,P. *et al.* (2010) Inferring cancer subnetwork markers using density-constrained biclustering. *Bioinformatics*, **26**, i625–i631.

Elbassioni,K. *et al.* (2012) The relation of Connected Set Cover and Group Steiner Tree. *Theor. Comput. Sci.*, **438**, 96–101.

Firsov,D. *et al.* (1996) Cell surface expression of the epithelial Na channel and a mutant causing Liddle syndrome: a quantitative approach. *Proc. Natl Acad. Sci. USA*, **93**, 15370–15375.

Jenssen,T.K. *et al.* (2001) A literature network of human genes for high-throughput analysis of gene expression. *Nat. Genet.*, **28**, 21–28.

Karp,R. (1972) Reducibility among combinatorial problems. In: Miller,R. and Thatcher,J (eds) *Complexity of Computer Computations*. Plenum Press, New York, pp. 85–103.

Ku,C.S. *et al.* (2011) Revisiting Mendelian disorders through exome sequencing. *Hum. Genet.*, **129**, 351–370.

Lee,I. *et al.* (2011) Prioritizing candidate disease genes by network-based boosting of genome-wide association data. *Genome Res.*, **21**, 1109–1121.

Lehne,B. and Schlitt,T. (2009) Protein-protein interaction databases: keeping up with growing interactomes. *Hum Genomics*, **3**, 291–297.

Lehne,B. and Schlitt,T. (2012) Breaking free from the chains of pathway annotation: de novo pathway discovery for the analysis of disease processes. *Pharmacogenomics*, **13**, 1967–1978.

McClellan,J. and King,M.C. (2010) Genetic heterogeneity in human disease. *Cell*, **141**, 210–217.

Mendig,A. *et al.* (2009) GPU-beschleunigtes 3D-Layout komplexer Netzwerke. In: von Lukas,U., *et al.* (eds) *Go-3D 2009: Go for Innovations*. Fraunhofer-Verlag, Stuttgart.

Ng,S.B. *et al.* (2009) Targeted capture and massively parallel sequencing of 12 human exomes. *Nature*, **461**, 272–276.

Ng,S.B. *et al.* (2010a) Exome sequencing identifies MLL2 mutations as a cause of Kabuki syndrome. *Nat. Genet.*, **42**, 790–793.

Ng,S.B. *et al.* (2010b) Massively parallel sequencing and rare disease. *Hum. Mol. Genet.*, **19**, R119–R124.

Ren,W. and Zhao,Q. (2011) A note on 'Algorithms for connected set cover problem and fault-tolerant connected set cover problem'. *Theor. Comput. Sci.*, **412**, 6451–6454.

Rossin,E.J. *et al.* (2011) Proteins encoded in genomic regions associated with immune-mediated disease physically interact and suggest underlying biology. *PLoS Genet.*, **7,** e1001273.

Simpson,M.A. *et al.* (2011) Mutations in NOTCH2 cause Hajdu-Cheney syndrome, a disorder of severe and progressive bone loss. *Nat. Genet.*, **43**, 303–305.

Staiger,C. *et al.* (2012) A critical evaluation of network and pathway-based classifiers for outcome prediction in breast cancer. *Plos One*, **7,** e34796.

Szklarczyk,D. *et al.* (2011) The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Res.*, **39**, D561–D568.

Ulitsky,I. *et al.* (2010) DEGAS: de novo discovery of dysregulated pathways in human diseases. *Plos One*, **5,** e13367.

Wang,B. *et al.* (2010) Gamma-secretase gene mutations in familial acne inversa. *Science*, **330**, 1065.

Warde-Farley,D. *et al.* (2010) The GeneMANIA prediction server: biological network integration for gene prioritization and predicting gene function. *Nucleic Acids Res.*, **38**, W214–W220.

Xu,K. *et al.* (2011) Path lengths in protein-protein interaction networks and biological complexity. *Proteomics*, **11**, 1857–1867.

Zhang,Z. *et al.* (2009) Algorithms for connected set cover problem and fault-tolerant connected set cover problem. *Theor. Comput. Sci.*, **410**, 812–817.