Bioinformatics, 32(7), 2016, 1001–1008
doi: 10.1093/bioinformatics/btv510
Advance Access Publication Date: 29 August 2015
Original Paper

OXFORD

Sequence analysis

Utilizing de Bruijn graph of metagenome assembly for metatranscriptome analysis

Yuzhen Ye* and Haixu Tang

School of Informatics and Computing, Indiana University, Bloomington, IN 47405, USA

*To whom correspondence should be addressed.

Associate Editor: Jan Korbel

Received on March 10, 2015; revised on June 3, 2015; accepted on August 24, 2015

Abstract

Motivation: Metagenomics research has accelerated the studies of microbial organisms, providing insights into the composition and potential functionality of various microbial communities. Metatranscriptomics (studies of the transcripts from a mixture of microbial species) and other meta-omics approaches hold even greater promise for providing additional insights into functional and regulatory characteristics of the microbial communities. Current metatranscriptomics projects are often carried out without matched metagenomic datasets (of the same microbial communities). For the projects that produce both metatranscriptomic and metagenomic datasets, their analyses are often not integrated. Metagenome assemblies are far from perfect, partially explaining why metagenome assemblies are not used for the analysis of metatranscriptomic datasets.

Results: Here, we report a reads mapping algorithm for mapping of short reads onto a de Bruijn graph of assemblies. A hash table of junction *k*-mers (*k*-mers spanning branching structures in the de Bruijn graph) is used to facilitate fast mapping of reads to the graph. We developed an application of this mapping algorithm: a reference-based approach to metatranscriptome assembly using graphs of metagenome assembly as the reference. Our results show that this new approach (called TAG) helps to assemble substantially more transcripts that otherwise would have been missed or truncated because of the fragmented nature of the reference metagenome.

Availability and implementation: TAG was implemented in C++ and has been tested extensively on the Linux platform. It is available for download as open source at http://omics.informatics.indiana.edu/TAG.

Contact: yye@indiana.edu

1 Introduction

Metagenomes are being generated at an accelerating pace, revealing important properties of microbiomes. Other meta-omic (e.g. metatranscriptomic and metaproteomic) techniques can provide additional insights, in particular into functional characteristics of microbial communities, such as gene activities and their regulatory mechanisms. Bacteria have low inventories of short-lived mRNAs; as such, fluctuations in their mRNA pools provide a highly sensitive bioassay for environmental signals [e.g. the concentrations of dissolved organic carbon (Shi *et al.*, 2012) and pollutant concentrations (de Menezes *et al.*, 2012) relevant to microbes (Moran *et al.*, 2013)]. The acquisition of meta-omics data on human microbiomes will enable us to refine the annotations of the metagenomes [the ENCODE (Dunham *et al.*, 2012) and modENCODE (Roy *et al.*, 2010) projects are great exemplars], and more importantly to study gene activity and its regulation (Maurice *et al.*, 2013) in complex microbial communities in order to understand how microbial organisms work as a community in response to changes in their environment, e.g. health conditions of their human hosts (Jorth *et al.*, 2014). A recent metatranscriptomic study of the human oral microbiome using patient-matched healthy and diseased (periodontal) samples revealed that health- and disease-associated communities exhibit defined differences in metabolism that are conserved between patients while the metabolic gene

expression of individual species was highly variable between patients (Jorth *et al.*, 2014).

In a metatranscriptomic RNA-seq study, total RNA is first isolated from the sample (with rRNAs removed to enrich for mRNA), which is then reverse transcribed into cDNA, and subjected to sequencing using next-generation sequencing platforms (Gosalbes *et al.*, 2011). Unlike metagenomics, which reveals potential activity (as reflected in genes or pathways that can be coded for by metagenomic sequences), metatranscriptomic data indicate which of the genes/metabolic pathways are actually active (and the level of their activities) on the basis of their transcription within the community. Giannoukos *et al.* (2012) presented a protocol for metatranscriptomic analysis of bacterial communities that accommodates both intact and fragmented RNA and combines efficient rRNA removal with strand-specific RNA-seq. Currently, only a handful of metatranscriptomic datasets are available (and metaproteomic datasets are even scarcer), but we envision a flood of metatranscriptomic data in the near future, as experimental techniques mature (Franzosa *et al.*, 2014; Giannoukos *et al.*, 2012).

Metatranscriptome analyses typically include the assignment of the predicted function and taxonomic origin of RNA-seq reads, by directly searching metatranscriptomic sequences (bags of reads) against prokaryotic genomes (the reference genomes) (Leimena *et al.*, 2013) or known protein sequences (Franzosa *et al.*, 2014). This way, tools and pipelines—including MG-RAST (Meyer *et al.*, 2008), MEGAN (Huson *et al.*, 2011) and HUMAnN (Abubucker *et al.*, 2012)—that have been developed for metagenome data analysis can be utilized for analyzing metatranscriptomic datasets. For example, Franzosa *et al.* (2014) analysed metagenomic and metatranscriptomic datasets of human gut microbiomes using the HUMAnN pipeline, revealing that metatranscriptional profiles were significantly more individualized than DNA-level functional profiles. One potential pitfall of such approaches is that they cannot identify transcripts of new genes, which however may be better annotated using assembly approaches (*de novo* or reference based). A recent study (Celaj *et al.*, 2014) compared the performances of currently employed transcriptome assemblers—including Trinity (Grabherr *et al.*, 2011), Oases (Schulz *et al.*, 2012a), Metavelvet (Namiki *et al.*, 2012) and IDBA-MT (Leung *et al.*, 2013)—and showed that assembly helps to improve the rate of functional annotation for metatranscriptomic datasets.

A matched metagenome can be helpful for the analysis of metatranscriptomic dataset. Metagenomes are often represented as contigs and scaffolds (although de Bruijn graphs are often the underlying data structure of the assemblers that were used), and are fragmented, limiting the utilization of metagenome for metatranscriptome analysis. There are pros and cons with the contig (and scaffold) representations of metagenomes. Most existing computational tools for sequence analysis work with linear representations of assemblies, so these tools (or modified versions) can be employed to analyse these representations of metagenomes. However, metagenomes are often very fragmented, and the connections between contigs or scaffolds are not captured in linear representations, which otherwise could be utilized later. For example, after we assembled two metagenomic datasets of stool samples from the Human Microbiome Project (Huttenhower *et al.*, 2012), the total lengths of scaffolds and contigs ($\geq$300 bp) reported by SOAPdenovo2 (Luo *et al.*, 2012) were about 85 and 90 Mb, respectively, whereas the total length of the edge sequences in the de Bruijn graph from the same assembly was 150 Mb for each. This comparison indicates that the de Bruijn graph representation of the assembly contains 50% more sequences than scaffolds reported from the assembler: most of these extra sequences are relatively fragmented sequences connecting long contigs. Furthermore, many short contigs contain only gene fragments; even long contigs contain broken genes at their ends due to the complexity of metagenome assembly (Wu *et al.*, 2012b).

Here, we propose a novel application of de Bruijn graphs for metatranscriptomic data analyses, taking advantage of the fact that de Bruijn graph representations of metagenome assemblies contain more information than the contigs and scaffolds reported by assemblers. The de Bruijn graph was first proposed for *de novo* genome assembly in EULER, replacing the traversal of Hamiltonian paths in the overlap graph by the traversal of Eulerian paths (Pevzner *et al.*, 2001), and is now employed as an efficient data structure in most short-read assemblers [e.g. Velvet (Zerbino and Birney, 2008), ALLPATHS-LG (Gnerre *et al.*, 2011), SOAPdenovo (Li *et al.*, 2010) and IDBA-UD (Peng *et al.*, 2012)] for single genomes and metagenomes. Our approaches based on de Bruijn graph representation of metagenomes provide a natural way of compressing the data, and, more importantly, allow direct utilization of the graphs. We note that we have developed several applications previously, based on de Bruijn graph representation of genomes and metagenomes, for mining of functional elements (Wu *et al.*, 2012a) and reads mapping (Wang *et al.*, 2012), demonstrating the utility of direct computation on de Bruijn graphs. Application of our method to simulated and real metatranscriptomic datasets showed that our approach can significantly improve the assembly of metatranscriptomic datasets, resulting in substantially more transcripts that otherwise would have been missed or truncated because of the fragmented nature of the reference metagenome.

2 Methods

In this article, we propose a novel algorithm (i.e. *read2graph*) for aligning short reads from RNA-seq experiments to de Bruijn graphs of assemblies. We note in this article we focused on de Bruijn graphs of metagenome assemblies, but the mapping algorithm can be applied to mapping short reads to any de Bruijn graph of assembly. We also developed an application of the mapping algorithm for metatranscriptome assembly using matched metagenomes as the reference. Based on reads mapping results, we will derive putative transcripts (encoding a single bacterial gene or multiple genes within an operon), using paired-end RNA-seq reads to traverse the de Bruijn graph. We named our transcript assembly approach TAG, in which TA stands for Transcript Assembly, and G is used to emphasize the fact that our approach utilizes the graph of metagenome assembly instead of the linear sequences. We note that our method is different from the *de novo* approaches to transcriptome assembly, including Trinity (Grabherr *et al.*, 2011), IDBA-MT (Leung *et al.*, 2013) [and also a hybrid approach (Leung *et al.*, 2014) that utilizes known protein sequences], and that it is different from the traditional reference-based assembly approaches. In our method, metatranscriptomic sequences are mapped onto matched metagenomes represented as de Bruijn graphs. So our method represents a new variant of the reference-based approaches, which uses the de Bruijn graph of matched metagenome, instead of a genome (or a collection of genomes), as the reference.

2.1 Fast reads mapping onto de Bruijn graph using a hash table of *k*-mers spanning branching structure in the graph

Given a de Bruijn graph, more exactly, a *contracted* de Bruijn graph (Cazaux *et al.*, 2014; Chang *et al.*, 2015), in which each edge

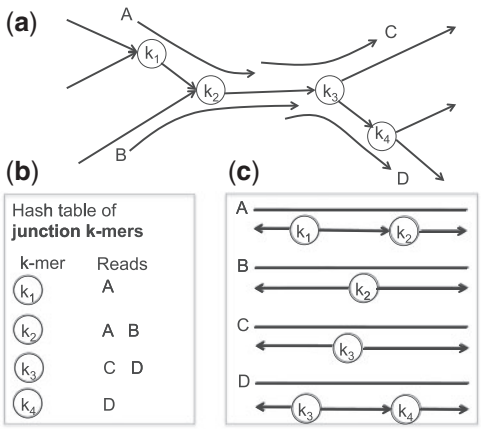

Fig. 1. A schematic illustration of the algorithm for mapping reads onto de Bruijn graphs. (**a**) A toy example showing four reads spanning *junction k-mers* in the graph (shown as the vertices). (**b**) Using a hash table of junction *k*-mers, candidates of reads that span multiple edges can be retrieved by looking up in the table. (**c**) For each candidate, a matched *k*-mer determines a unique putative location of the read in the graph (i.e. a seed match). The seed match will then be used to constrain the alignment between the read and the graph by a dynamic programming algorithm

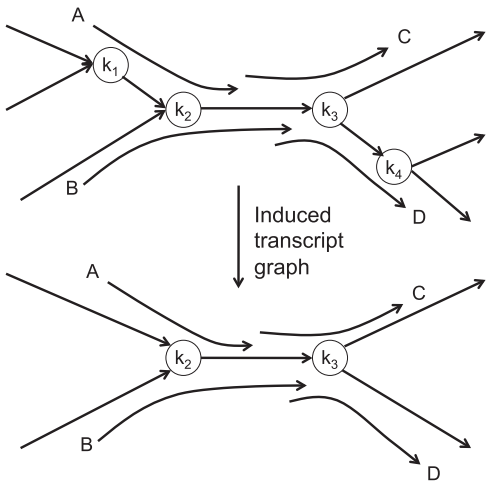

Fig. 2. A schematic example illustrating the induced transcript graph derived from four reads (A–D) mapped to a de Bruijn graph of metagenome assembly

represents an assembled unique sequence from metagenomic reads, and a set of short reads from an RNA-seq experiment, the goal of our read2graph algorithm is to find the location of each read on the graph. Because bacterial genes do not have split gene structure, we can assume each read should be contained in the graph as a whole; equivalently, each read, if its location in the graph is known, can be represented as a path (i.e. sequence of edges) in the graph. The reads, therefore, can be classified into two groups depending on the path length: some reads are located within a single edge, whereas many others may cross one or more vertices in the graph. The first class of reads can be mapped to the graph using conventional fast reads mapping algorithms by using all edge sequences longer than the read length as the target sequences. In this article, we used Bowtie 2 (Langmead and Salzberg, 2012) for this purpose; but other mapping algorithms including BWA (Li and Durbin, 2009) can be used. Here, we focus on the methods for mapping reads spanning multiple edges (i.e. *multi-edge-spanning reads*; see Fig. 1), which cannot be mapped using conventional mapping algorithms. A substantial number of reads may belong to this class, due to the incompleteness of metagenome assembly.

Recall that each vertex in the de Bruijn graph represents a *k*-mer in metagenomic reads [typically $k = 23$–31 for metagenome assembly (Huttenhower *et al.*, 2012; Qin *et al.*, 2010)]. Therefore, as illustrated in Figure 1, each multi-edge-spanning read contains one or more junction *k*-mers (i.e. corresponding to vertices with either indegree or outdegree >1): reads A and D span three edges in the graph, and thus each contains two such *k*-mers, whereas reads B and C span two edges, and thus each contains one such *k*-mer (Fig. 2a). Hence, we can build a hash table for all *junction k-mers* that span branching structures in the de Bruijn graph assembly and then search for their exact occurrences in each putative multi-edge-spanning read (i.e., those that cannot be mapped to the edge sequences) with the assistance of the hash table (Fig. 1b). Because each *k*-mer in the de Bruijn graph is unique (Pevzner *et al.*, 2001), every *k*-mer in a read matches at most one *k*-mer stored in the hash table. Each matched *k*-mer determines a unique putative location of the multi-edge-spanning read in the graph (i.e. a seed match between the read and the graph), and simultaneously breaks the read into two or more segments (Fig. 1c). The seed match will then be used to constrain the alignment between the read and the graph, starting from the seed match, going in opposite directions, using a constrained dynamic programming algorithm allowing only a small number of indels and mismatches. The bandwidth for constrained alignment is set to 7 by default for metatranscriptome assembly using matched metagenome as the reference, and this parameter can be changed by users for other purposes.

The mapping of multi-edge-spanning reads should run fast and consume reasonable memory because usually there are only hundreds of thousands of junction *k*-mers in a typical metagenome assembly in practice. We note that the multi-edge-spanning reads considered here are different from the split reads considered in transcript assembly for eukaryotes (Grabherr *et al.*, 2011), and in rare cases for archaeal species (due to tRNA splicing and self-splicing introns) (Doose *et al.*, 2013). Since strand-specific RNA-seq protocols are often used in metatranscriptome analysis (Giannoukos *et al.*, 2012), our algorithm can consider the strand information and map reads to one appropriate strand in the de Bruijn graph that contains sequences from both DNA strands (and thus is symmetric).

2.2 Construction of transcripts from mapped reads

Once all RNA-seq reads including multi-edge-spanning reads are mapped to the graph, each read can be represented by a path (referred to as the *read path*) traversing the graph $< e_1, e_2, \ldots, e_l >$ ($e_1, e_2, \ldots, e_l$ are edges; for non-multi-edge-spanning reads, path length $l = 1$) as well as two offset values representing the locations of the read in the first and last edges in the graph. Furthermore, in most cases, two paired-end reads can also be represented as a path (i.e. the *read-pair path*) if there exists a unique path in the graph whose length is consistent with the expected insert size. As a result, the assembly of RNA-seq reads is equivalent to the superpath problem, which attempts to find a minimal set of superpaths (each corresponding to a transcript) that covers a given set of paths in a de Bruijn graph (Nagarajan and Pop, 2009). Although this problem is generally hard, we can represent the solutions of the problem in a much simpler subgraph (the transcript graph) that contains only the edges present in at least one of the read paths or read-pair paths. Figure 2 shows such an example: assuming four read paths (A, B, C and D) are derived from multi-edge-spanning reads, we will induce the transcript graph by retaining all edges in these paths, and then

contracting all vertices with both indegree and outdegree of 1. We note that many read paths may contract into a single edge in the transcript graph if they are not tangled with reads from another transcript (e.g. k1 and k4 in Fig. 2 can be contracted because there are no conflicting transcript reads traversing through these nodes); as a result, the corresponding transcript sequences can be retrieved as a subsequence of an edge sequence in the transcript graph. In other cases, read paths remain spanning multiple edges in the transcript graph. These read paths sometimes can be used to further simplify transcript graph, as illustrated in the heuristic algorithms in genome assembly (Pevzner *et al.*, 2001; Zerbino and Birney, 2008). For instance, in the example shown in Figure 2, if we have two read-pair paths spanning AC and BD, respectively, we can obtain two resolved transcripts from the graph. Otherwise, we can only obtain partial transcript sequences. We note that, even if the transcripts cannot be fully resolved, the transcript graph is still useful for inferring the abundances of putative transcripts in a metatranscriptome sample based on the counts of reads on the edges in the transcript graph, a problem similar to the inference of splicing variants in eukaryotic RNA-seq experiments (see Pachter, 2011 for a review).

2.3 Metatranscriptome assembly using metagenome assembly graph as the reference

Our approach for metatranscriptome assembly (called TAG) is based on the read2graph mapping algorithm and the transcript construction approach as described above. Given a metatranscriptomic dataset and a matched metagenomic dataset, SOAPdenovo2 (Luo *et al.*, 2012), one commonly used assembler in metagenomic shotgun sequencing, is used to assemble the metagenomic dataset. Notably, SOAPdenovo2 is a de Bruijn graph-based assembler, and in its final output, both the de Bruijn assembly graph and the contig sequences (representing the *edges* in the graph) are produced. The mapping of metatranscriptomic sequences to the de Bruijn graph is conducted in two consecutive steps: (1) all reads are first mapped to the edges (i.e. contigs) in the de Bruijn graph using Bowtie 2 (version 2.2.3) (Langmead and Salzberg, 2012), and then, (2) the un-mapped reads in the previous step are further mapped to the graph based on the matching with junction k-mers. Next, TAG traverses the de Bruijn graph along with the mapped metatranscriptomic reads, and reports the transcripts that may span multiple edges in the assembly graph. To use the strand-specific information, the mapping of a metatranscriptomic sequence is only considered for the strand of the read that represents the transcript (i.e. the forward strand of R2 reads and the reverse-complement strand of R1 reads for the datasets we have tested). We note that other short read assemblers [such as IDBA (Peng *et al.*, 2012)] and mapping tools [such as BWA (Li and Durbin, 2009)] can be utilized for generating the inputs (i.e. the de Bruijn assembly graph and the mapping of metatranscriptomic reads to contigs) for TAG. For the rest of this article, we will focus on the utility of TAG on improving the assembly of transcripts, which will be demonstrated by using the SOAPdenovo2 and Bowtie2 tools. The construction of an optimal pipeline (in particular the selection of upstream software tools) utilizing TAG is beyond the scope of this article.

3 Results

We tested our tool (TAG) on two metatranscriptomic datasets (Giannoukos *et al.*, 2012): one derived from a mock microbial community consisting of three bacterial species, and the other derived from a real microbiome sample in human stool. Results showed that our graph-based reads mapping algorithm (read2graph) is efficient, and TAG, which is based on the mapping algorithm, significantly improves the assembly of metatranscriptomes by considering reads mapping to branching structures in de Bruijn graphs of matched metagenomes.

3.1 Evaluation of assembly accuracy on a mock dataset

We first tested TAG using a metranscriptomic data from the mock bacterial community of three species (Giannoukos *et al.*, 2012). The 'matched' metagenomic dataset used in TAG were simulated from the reference genomes of these bacteria [*Escherichia coli* (GenBank: NC_000913.3), *Perkinsus marinus* (GenBank: NC_005072.1) and *Rhodobacter sphaeroides* (GenBank: NC_007493.2)] using NeSSM (Jia *et al.*, 2013) with the Illumina error model. We used this *hybrid* approach here because (1) there is currently no metatranscriptomic dataset from a mock community with a matched metagenomic dataset available, and (2) there is no proper software tool for simulating metatranscriptomic dataset. (Flex Simulator is a tool for simulating RNAseq data for single species, and it has been mainly used for eukaryotic species. Bacteria have complicated transcription regulation mechanisms, which are not completely understood.) In total, 1 M paired-end reads of length 101 bp (i.e. $\sim 20\times$ coverage) were simulated from the three species with equal abundances. SOAPdenov2 (version 2.04-r240) ($k = 31$; see below for the choice of k-mer size) was used to assemble the simulated reads, and the assembly results (including the contigs and the de Bruijn assembly graph) were then used as the inputs to TAG. Because this is a simple community with bacterial species that are phylogenetically distant (Giannoukos *et al.*, 2012), the assembly graph of the metagenome is not very tangled, and thus we do not anticipate that many transcripts reported by TAG will span multiple edges (referred to as the *multi-edge transcripts*) in the assembly graph. In fact, TAG reported a total of 9428 transcripts (of $\geq$100 bp), among which only 138 are multi-edge spanning transcripts.

3.1.1 Accuracy evaluation for the TAG transcripts

We blasted transcripts assembled by TAG against the three reference genomes to evaluate the accuracy of metatranscriptome assembly. Our results showed that only 16 out of 9428 (0.17%) transcripts cannot be perfectly aligned back to the reference genomes: among the 16 transcripts, 14 can be aligned with minor differences, and only two contain potentially serious problems (see Table 1). We note that there are two types of potential errors in the transcripts assembled by TAG: the errors introduced by TAG, and the errors inherited from the metagenome assembly (i.e. the mis-assemblies present in the metagenome assembly that propagates into the transcript). One of the problematic transcript is single-edge transcript, suggesting that this assembly error was propagated from the metagenome assembly. The other problematic transcript (of 390 bp) is a multi-edge spanning transcript, and the error was introduced by TAG (as no matching sequence can be found in the metagenome assembly). Our results suggest that TAG achieves high assembly accuracy overall with an error rate of $<<1\%$. If we only focused on multi-edge spanning transcripts (which are more difficult to assemble than transcripts contained within edges and therefore more error prone), the assembly error rate is still very low: only one out of 138 multi-edge transcripts contains such large assembly problem (the error rate is 0.7%).

Table 1. Performance comparison of TAG and other assemblers on the mock dataset

	Oases	Trinity	TAG
No. of transcripts[a]	12598	24804[b]	9428
Perfectly aligned transcripts (percentage)[c]	5483 (43.5%)	12392 (50.0%)	9412 (99.8%)
Transcripts with minor problems (percentage)[c]	2724 (21.6%)	2725 (11.0%)	14 (0.15%)
Problematic transcripts (percentage)[c]	4391 (34.9%)[d]	9687 (39.1%)[d]	2 (0.02%)
Total length of the transcripts	6860841 bp	7428187 bp	7020975 bp
Total length of perfectly aligned transcripts	2265224 bp	3858486 bp	7002290 bp
Total length of good transcripts	4076481 bp	5025072 bp	7020484 bp

[a]Only transcripts of at least 100 bp were considered for all programs.

[b]Trinity has many more transcripts, but their total length is comparable to the other methods.

[c]A transcript that is perfectly aligned to one of the reference genomes (with an alignment covering the entire transcript at 100% sequence identity) is considered to be correctly assembled. We consider the problem of a transcript is 'minor' if its longest alignment with the reference genomes is not 10 nt shorter than the transcript and the alignment has 95% sequence identity or better. Other transcripts that do not meet these criteria are considered to be problematic.

[d]A large fraction of the problematic transcripts for Oases and Trinity are likely caused by the presence of contaminated sequences or other artifacts so should not be considered as mis-assemblies. For example, 3494 (out of 4391) Oases transcripts have no significant alignments with the reference genomes with *E*-values better than 1e − 4, and therefore are unlikely transcripts from the reference genomes.

3.1.2 Comparison with de novo assembly

We further compared the performance of TAG with Oases (version 1.2.10) (Schulz *et al.*, 2012a) and Trinity (release 2014-07-17) (Grabherr *et al.*, 2011), *de novo* assemblers for transcriptomic sequences. (**Trinity has been applied to analyse metatranscriptomic datasets (Celaj *et al.*, 2014), although the program was developed targeting splicing isoforms in Eukaryotes.**) For Oases, we used merged results from assemblies using *k*-mer sizes ranging from 19 to 31. Table 1 summarizes the comparison results. Although Oases and Trinity produced larger numbers of transcripts than TAG, the total bases in the transcripts assembled by these three methods are comparable (i.e. TAG assembled longer transcripts). If we considered only the 'good' transcripts by excluding the transcripts that cannot be aligned well to the reference genomes [which are likely misassembles, or assemblies from contaminated sequences or other artifacts commonly found in RNA-seq experiments (Lahens *et al.*, 2014)], the difference in the total lengths of transcripts is even more significant. TAG produced a total of 9426 good transcripts with a total of 7020484 bp, while Oases and Trinity assembled transcripts of 407648 and 5025072 total bases, respectively. This result shows that using reference genomes for metatranscriptome assembly helps to improve the coverage and quality of the assemblies.

We ran CD-HIT-EST (version 4.6) (Li and Godzik, 2006) to cluster the good transcripts from all programs at 95% sequence identity cutoff (−c 0.95), resulting in 10 944 clusters: only a modest number of clusters (2309) are shared by all methods, 2965 clusters are shared by two methods (1399 shared Trinity and Oases; 1369 by Trinity and TAG; and 116 by TAG and Oases), and the remaining clusters are unique to one method (TAG: 2571, Trinity: 2983, and Oases: 197). We quantified the abundances of the transcripts by mapping metatranscriptomic sequences onto the transcripts using Bowtie2. The transcripts that are shared by all methods are highly abundant with an average coverage of 28.2 (i.e. on average, each position is covered by 28.2 reads). In contrast, the average abundances of the transcripts that can be assembled by Oases, Trinity and TAG are 11.4, 8.0 and 6.4, respectively. This result suggests that *de novo* assembly and reference-based approaches can complement each other: transcripts of highly expressed genes in rare species (and therefore less well represented in metagenomes) may be assembled by *de novo* assembly, while transcripts of low expression level can be better identified using reference-based approaches.

3.2 Application of TAG to a real metatranscriptomic dataset

We applied TAG to analysing a metatranscriptomic dataset derived from a human stool sample, using its matched metagenomic dataset as the reference (Giannoukos *et al.*, 2012). (We combined the metatranscriptomic reads from four fractions of sequencing of the same sample, downloaded from SRA (SRX130930, SRX130937, SRX130922 and SRX130928), and the metagenomic reads from four fractions of sequencing, also downloaded from SRA (SRX130930, SRX130954, SRX130936 and SRX130949). Note that we used the metatranscriptomic dataset sequenced on 5 μg RNA extracted from an individual's stool microbiome, which was shown to yield the best sequencing results (Giannoukos *et al.*, 2012).) As described above, the metagenomic sequences were first assembled using SOAPdenovo2, and the metagenome assembly was then used as the reference for the metatranscriptome assembly by TAG.

3.2.1 Time and memory cost of the reads mapping to the de Bruijn graph

Metatranscriptome assembly by TAG (including reads mapping onto the graph and the transcript inference afterwards) for this dataset takes about 7 min to complete on a Linux computer with Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80 GHz (using single processor). The actual reads mapping step takes about 1 min to complete—the remaining 6 min were spent on other I/O steps including processing the input SAM alignment file (from Bowtie2) and reads files. This indicates that our graph-based reads mapping algorithm (read2graph) is efficient. TAG adds only a small amount of computational time to the whole pipeline for the metatranscriptome analysis—SOAPdenovo2 takes several hours to assemble the metagenome, and mapping metatranscriptomic reads onto the metagenome contigs by Bowtie2 takes about 1700 CPU minutes (the actual job was done in parallel using 32 processors).

The memory consumption by TAG is bounded by the size of the input metagenome assembly (which is used as the reference). TAG consumed <2 GB RAM for the stool dataset. It shows another advantage of using metagenome assembly as the reference for metatranscriptome analysis, since de Bruijn graph provides a compact representation of the metagenome assembly (but still keeps the uncertainties of the assembly in the graph for future applications).

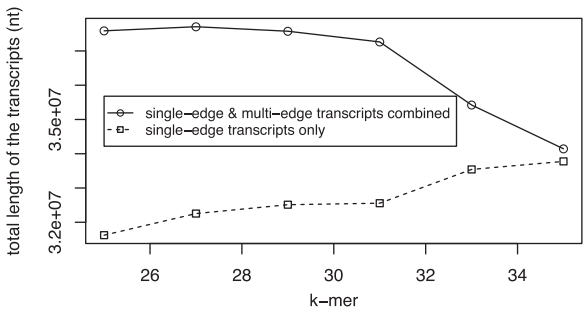

Fig. 3. The impact of *k*-mer size on the performance of TAG. When the *k*-mer size increases from 25 to 31 in SOAPdenovo2 assembly, the performance of TAG remains the same: a substantial fraction of multi-edge transcripts can be assembled by TAG. However, when further increasing the *k*-mer size to 35, most transcripts assembled by TAG are single-edge transcripts, indicating the TAG algorithm is not effective when a large *k*-mer is used. This is probably because, in this case, the metagenome assembly is fragmented rather than tangled, and as a result the total length of the transcript also decreases. Therefore, in the experiments of this article, we choose $k=31$ in SOAPdenovo2 assembly, which seems to yield the best results here

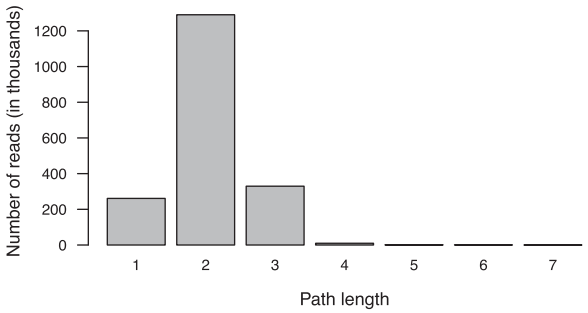

Fig. 4. The path length distribution for *multi-edge-spanning reads* that span two or more edges when mapped to the de Bruijn graph by TAG. The *X*-axis represents the length of multi-edge-spanning read paths (i.e. the number of edges that the multi-edge-spanning reads span) and the *Y*-axis represents the total number of multi-edge-spanning reads spanning the paths of certain lengths. Paths of length 1 represent the cases when the seed extension in one direction resulted in an alignment of at most 7 bp, and thus were considered insignificant and discarded

3.2.2 Exploiting tangles in de Bruijn graph to improve metatranscriptome assembly

We tested the performance of TAG using reference metagenomes assembled with different *k*-mer sizes, considering that the choice of *k*-mer size is important for the metagenome assembly (Li *et al.*, 2010; Zerbino and Birney, 2008) and therefore metatranscriptome assembly. As shown in Figure 3, when a relatively small *k*-mer (e.g. 25) was used, the metagenome assemblies are more tangled, and as a result, fewer transcripts can be assembled using the contigs as the reference. This pitfall, however, can be alleviated by retaining the tangled structure (i.e. the ambiguous connection caused by short repeats) in the metagenome assembly in the de Bruijn graph, which can be exploited by TAG to connect metatranscriptomic reads into complete transcripts, resulting in improved assembly of metatranscriptome.

As shown in Figure 3, the total length of the assembled transcripts by TAG decreases slowly when *k*-mer size increases from 25 to 31. Considering that most transcripts are longer at *k*-mer = 31 as compared with smaller *k*-mers (e.g. average lengths of the transcripts are 264 and 273 for *k*-mer = 25 and 31, respectively), we selected *k*-mer = 31 to demonstrate the improvement of metatranscriptome assembly by using TAG. Figure 4 shows the distribution of the path lengths (i.e. the number of edges that are traversed in the de Bruijn graph to form a transcript by TAG) of the transcripts assembled by TAG: most of the multi-edge transcripts span two edges (contigs), although a single transcript may span as many as seven edges.

Table 2. Some statistics of TAG assembly on the human stool metatranscriptomics dataset

Total number of reads	27962127×2 (paired)
Number of reads mapped to contigs	$19233474 \times 2 + 7645742$ (single)
Number of multi-edge-spanning reads	1893157
Number of *resolved*[a] single-edge transcripts (length)	112527 (32216351 bp)
Number of *partial* [a] single-edge transcripts (length)	2573 (340276 bp)
Total number of *single-edge*[b] transcripts (length)	115100 (32556627 bp)
Number of *resolved* of multi-edge transcripts (length)	20903 (4596622 bp)
Number of *partial* multi-edge transcripts (length)	552 (110063 bp)
Total number of *multi-edge*[b] (length) transcripts (length)	21455 (4706685 bp)
Total number of transcripts (length)	177463 (40456052 bp)
Proportion of multi-edge transcripts (in length)	15.7% (11.6%)

Only transcripts of at least 100 bp were considered in this summary.

[a]Partial transcripts: the transcripts that are not fully resolved by TAG (i.e. the edge sequences); Resolved transcripts: the transcripts that are resolved by TAG and therefore likely represent full-length transcripts.

[b]Single-edge transcripts: the transcripts reported by TAG that are fully contained within edges (contig) in the de Bruijn graph of the metagenome assembly (they can be considered as the results of a baseline reference-based metatranscriptome assembly approach that uses the contigs as the reference); Multi-edge transcripts: the transcripts reported by TAG that span multiple edges in the de Bruijn graph.

Table 2 summarizes the metatranscriptome assembly results by TAG. A majority of the metagenomic reads can be mapped to metagenomic assembly: for 68.8% of read pairs, both reads can be mapped to contigs by Bowtie2, whereas an additional 13.6% reads can be mapped to contigs although their mate-pairs cannot be mapped. Among the $\approx 9.8M$ remaining unmapped reads, $\approx 1.9M$ (18.9%) can be mapped to multiple edges (i.e. through one or more junction *k*-mers in the de Bruijn graph) by TAG. Thanks to these reads, TAG was able to improve the metagenomic assembly significantly. In total, TAG assembled about 177K transcripts, among which about 21K (15.7%) are multi-edge transcripts. These multi-edge transcripts cannot be fully assembled if only those reads mapped to contigs are considered in the metatranscriptome assembly; instead, they are likely to be broken into *partial* transcripts, each contained in a separate contig (i.e. the edge in the de Bruijn graph). We note that TAG did not resolve all transcripts. A small fraction of TAG-assembled transcripts are *partial* transcripts, each of which represents a unique edge in the tangled *transcript graph*, formed by two or more transcripts sharing some common segments (see Section 2 for details) that cannot be resolved without additional information. About 2.6% (552 out of 21455) of the multi-edge

transcripts were not fully resolved by TAG and remained as partial transcripts. Similarly, 2.2% (2573 out of 115100) of the single-edge transcripts are also partial transcript as some multi-edge-spanning reads connect them with other partial transcripts, although their actual connections remain ambiguous. We note that these two numbers increase substantially (to 21.1 and 8.1%, respectively) when there is no minimum length applied for output transcripts.

We also compared the TAG assemblies with the *de novo* transcript assemblies from Trinity. We note that this is a real metatranscriptomic dataset, so that we cannot compare the results in terms of the accuracy of the assembly as we did for the mock dataset (but we have shown using the mock dataset that *de novo* assembly tends to produce more problematic transcripts). In total, TAG produced 136 555 transcripts with a total of 37.4 Mb, whereas Trinity generated 207697 transcripts with a total of 44.8 Mb. Similar to the results on the mock dataset, TAG transcripts are longer than Trinity transcripts: the average lengths of the transcripts are 273 and 216 bp, for TAG and Trinity, respectively. Combining the transcripts from both assemblers (and removing redundant transcripts at 95% sequence identity by CD-HIT-EST) resulted in 233 201 transcripts with a total of ~55.8 Mb, again demonstrating that reference-based and *de novo* approaches can complement with each other to improve the coverage of transcript assembly.

4 Discussion

Even though thousands of complete prokaryotic genomes and many more draft genomes are available, metagenomes are constantly found to contain many new species and new genes (Huttenhower *et al.*, 2012; Qin *et al.*, 2010; Vital *et al.*, 2014). It is therefore important to develop methodologies for metatranscriptome data analysis that are not constrained by the sequenced genomes. With 'matched' metagenomic and metatranscriptomic datasets, we believe that proper utilization of the metagenome data will help greatly the analysis of metatranscriptomic data (and *vice versa*). The eventual integration of these datasets (as well as other meta-omic datasets) will provide new insights on the composition, function and regulation of microbiomes. Well-assembled transcripts are important for the function annotation of the metatranscriptome, and also for inferring gene regulatory mechanisms such as the operons.

We developed a novel reads mapping algorithm (read2graph) that allows fast mapping of short reads from transcriptome sequencing onto the assembly graphs of reference genomes. We applied this mapping algorithm for metatranscriptome assembly, showing the utility of the de Bruijn assembly graph of the metagenome in downstream applications such as the metatranscriptome analysis. Our mapping tool is fast and can be applied to other applications, for example, mapping metagenomic sequencing reads onto the de Bruijn graph of closely related species for estimating the relative abundances of these species (Wang *et al.*, 2012). We have shown in a related research that genes are often broken into fragments in metagenome assembly, and multi-edge-spanning reads can stitch them together (Wu *et al.*, 2012b). The mapping of multi-edge-spanning reads will also improve quantification of gene expression based on read counts, in particular for genes (from the same or different organisms) sharing highly similar sequences. In reality, however, we may still miss the mapping of a small fraction of multi-edge-spanning reads: if a read contains a sequencing error in the occurrence of a branching k-mer, we cannot find its location in the graph. Because of the low error rate (<1%) in Illumina reads, we believe this fraction of reads is indeed negligible in metatranscriptomic data analysis.

We note that de Bruijn graphs will naturally capture the genomic variations of the metagenomes in the graphs [e.g. the single-nucleotide variations are represented as bulges (Nijkamp *et al.*, 2013), the variations in tandem repeats are represented as wheels, and structural variations are represented long loops (Pevzner *et al.*, 2001)], which is yet another advantage of using graphs instead of contigs to represent metagenomes. Genomic variations in metagenomes are naturally handled by our graph-centric mapping approach.

We expect that a combination of different approaches (reference-based and *de novo*) need to be applied to accomplish the comprehensive metatranscriptome analysis. As the references for metatranscriptome analysis, the matched metagenome will never be perfect, due to biological (rare species may be poorly sampled), experimental (some genomic regions may not be covered well) and computational (assemblers are not perfect) reasons. Integration of known reference genomes, matched metagenomes and even non-matched metagenomes can maximize the coverage of references for reference-based approaches. On the other hand, if a microbial community contains new, rare but highly expressed microbial species, their transcripts can only be revealed by *de novo* metatranscriptome assembly (Schulz *et al.*, 2012b) but not by the reference-based approaches such as the one presented in this article.

Funding

This research was supported by National Institutes of Health (NIH) grant 1R01AI108888-01A1.

Conflicts of Interest: none declared.

References

Abubucker,S. *et al.* (2012). Metabolic reconstruction for metagenomic data and its application to the human microbiome. *PLoS Comput. Biol.*, **8**, e1002358.

Cazaux,B. *et al.* (2014). From indexing data structures to de bruijn graphs. In:Kulikov, A., Kuznetsov, S. and Pevzner, P. (eds.), *Combinatorial Pattern Matching*, volume 8486 *of Lecture Notes in Computer Science*, pages 89–99. Springer International Publishing.

Celaj,A. *et al.* (2014). Comparison of assembly algorithms for improving rate of metatranscriptomic functional annotation. *Microbiome*, **2**, 39.

Chang,Z. *et al.* (2015). Bridger: a new framework for de novo transcriptome assembly using RNA-seq data. *Genome Biol.*, **16**, 30.

de Menezes,A. *et al.* (2012). Comparative metatranscriptomics reveals widespread community responses during phenanthrene degradation in soil. *Environ. Microbiol.*, **14**, 2577–2588.

Doose,G. *et al.* (2013). Mapping the RNA-Seq trash bin: unusual transcripts in prokaryotic transcriptome sequencing data. *RNA Biol.*, **10**, 1204–1210.

Dunham,I. *et al.* (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, **489**, 57–74.

Franzosa,E.A. *et al.* (2014). Relating the metatranscriptome and metagenome of the human gut. *Proc. Natl Acad. Sci. U. S. A.*, **111**, E2329–E2338.

Giannoukos,G. *et al.* (2012). Efficient and robust RNA-seq process for cultured bacteria and complex community transcriptomes. *Genome Biol.*, **13**, R23.

Gnerre,S. *et al.* (2011). High-quality draft assemblies of mammalian genomes from massively parallel sequence data. *Proc. Natl Acad. Sci. U. S. A.*, **108**, 1513–1518.

Gosalbes,M.J. *et al.* (2011). Metatranscriptomic approach to analyze the functional human gut microbiota. *PLoS One*, **6**, e17447.

Grabherr,M. *et al.* (2011). Trinity: reconstructing a full-length transcriptome without a genome from RNA-Seq data. *Nat. Biotechnol.*, **29**, 644–652.

Huson,D.H. *et al.* (2011). Integrative analysis of environmental sequences using MEGAN4. *Genome Res.*, **21**, 1552–1560.

Huttenhower,C. *et al.* (2012). Structure, function and diversity of the healthy human microbiome. *Nature*, **486**, 207–214.

Jia,B. *et al.* (2013). NeSSM: a Next-generation sequencing simulator for metagenomics. *PLoS One*, **8**, e75448.

Jorth,P. *et al.* (2014). Metatranscriptomics of the human oral microbiome during health and disease. *MBio*, **5**, e01012–e01014.

Lahens,N.F. *et al.* (2014). IVT-seq reveals extreme bias in RNA sequencing. *Genome Biol.*, **15**, R86.

Langmead,B. and Salzberg,S.L. (2012). Fast gapped-read alignment with Bowtie 2. *Nat. Methods*, **9**, 357–359.

Leimena,M.M. *et al.* (2013). A comprehensive metatranscriptome analysis pipeline and its validation using human small intestine microbiota datasets. *BMC Genomics*, **14**, 530.

Leung,H.C. *et al.* (2013). IDBA-MT: de novo assembler for metatranscriptomic data generated from next-generation sequencing technology. *J. Comput. Biol.*, **20**, 540–550.

Leung,H. *et al.* (2014). IDBA-MTP: A hybrid metatranscriptomic assembler based on protein information. *Res. Comput. Mol. Biol.. Lect. Notes Comput. Sci.*, **8394**, 160–172.

Li,H. and Durbin,R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, **25**, 1754–1760.

Li,W. and Godzik,A. (2006). Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, **22**, 1658–1659.

Li,R. *et al.* (2010). De novo assembly of human genomes with massively parallel short read sequencing. *Genome Res.*, **20**, 265–272.

Luo,R. *et al.* (2012). SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler. *Gigascience*, **1**, 18.

Maurice,C.F. *et al.* (2013). Xenobiotics shape the physiology and gene expression of the active human gut microbiome. *Cell*, **152**, 39–50.

Meyer,F. *et al.* (2008). The metagenomics RAST server—a public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC Bioinformatics*, **9**, 386.

Moran,M.A. *et al.* (2013). Sizing up metatranscriptomics. *ISME J.*, **7**, 237–243.

Nagarajan,N. and Pop,M. (2009). Parametric complexity of sequence assembly: theory and applications to next generation sequencing. *J. Comput. Biol.*, **16**, 897–908.

Namiki,T. *et al.* (2012). MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads. *Nucleic Acids Res.*, **40**, e155.

Nijkamp,J.F. *et al.* (2013). Exploring variation-aware contig graphs for (comparative) metagenomics using MaryGold. *Bioinformatics*, **29**, 2826–2834.

Pachter,L. (2011). Models for transcript quantification from rna-seq. *arXiv preprint arXiv:1104.3889*.

Peng,Y. *et al.* (2012). IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics*, **28**, 1420–1428.

Pevzner,P.A. *et al.* (2001). An Eulerian path approach to DNA fragment assembly. *Proc. Natl Acad. Sci. U. S. A.*, **98**, 9748–9753.

Qin,J. *et al.* (2010). A human gut microbial gene catalogue established by metagenomic sequencing. *Nature*, **464**, 59–65.

Roy,S. *et al.* (2010). Identification of functional elements and regulatory circuits by Drosophila modENCODE. *Science*, **330**, 1787–1797.

Schulz,M.H. *et al.* (2012a). Oases: robust de novo RNA-seq assembly across the dynamic range of expression levels. *Bioinformatics*, **28**, 1086–1092.

Schulz,M.H. *et al.* (2012b). Oases: robust de novo rna-seq assembly across the dynamic range of expression levels. *Bioinformatics*, **28**, 1086–1092.

Shi,Y. *et al.* (2012). Transcriptional responses of surface water marine microbial assemblages to deep-sea water amendment. *Environ. Microbiol.*, **14**, 191–206.

Vital,M. *et al.* (2014). Revealing the bacterial butyrate synthesis pathways by analyzing (meta)genomic data. *MBio*, **5**, e00889.

Wang,M. *et al.* (2012). A de Bruijn graph approach to the quantification of closely-related genomes in a microbial community. *J. Comput. Biol.*, **19**, 814–825.

Wu,Y.W. *et al.* (2012a). Oral spirochetes implicated in dental diseases are widespread in normal human subjects and carry extremely diverse integron gene cassettes. *Appl. Environ. Microbiol.*, **78**, 5288–5296.

Wu,Y.W. *et al.* (2012b). Stitching gene fragments with a network matching algorithm improves gene assembly for metagenomics. *Bioinformatics*, **28**, i363–i369.

Zerbino,D.R. and Birney,E. (2008). Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res.*, **18**, 821–829.