CAREER: ABI: Scaling and Improving de Bruijn graph assembly

C. Titus Brown

Monday, July 25th, 2011

Keywords: assembly, bioinformatics, next generation sequencing, metagenomics, transcriptomics, mRNAseq.

Project Summary

De novo assembly of genomic, transcriptomic, and metagenomic sequences without a reference is a critical endeavor in modern biology, in part because we have yet to sample even a small fraction of the tree of life. Yet sequencing capacity has recently begun to match our desire to investigate the natural world: we can now easily sequence entire genomes, transcriptomes, and metagenomes within a single lab [Tyson et al., 2004, Mortazavi et al., 2010, Li et al., 2010]. Moreover, the rate at which sequencing capacity is growing is itself increasing, leading to an *exponential* increase in sequencing capacity that is outstripping Moore's Law (www.genome.gov/sequencingcosts/).

De novo assembly techniques have not kept up with the advances in sequencing. A large class of modern assemblers, de Bruijn graph assemblers, has been developed for the express purpose of short-read assembly and can scale to assemble single human genomes on commodity hardware [Gnerre et al., 2011]. However, these assemblers are neither designed for nor scale to the volume of data being generated for transcriptomes or metagenomes, which can contain more novel sequence than genomic samples. Scaling transcriptome and metagenome assembly without compromising the quality of the assembly is one of the dominant bioinformatic problems of this decade.

Research Objectives: I therefore propose a research plan centered on **applying a novel *data structure*, a probabilistic de Bruijn graph, to the problem of scaling and improving de novo assembly of metagenomes and transcriptomes**. Our long term goal is to *improve assembly techniques overall* by scaling assembly approaches. I propose to use this novel data structure to develop algorithms and approaches for scaling and improving assembly in existing and emerging sequencing data sets, and to build practically useful tools implementing these approaches.

Intellectual merits: This project will contribute significantly to *biological understanding* of complex genomic samples by scaling assembly to larger, deeper samples, and improving assembly quality, which in turn will improve the foundation of many biological investigations. It will also provide a new set of *computational approaches* for understanding and taking advantage of assembly graph structures. Finally, we will help address the effects of next-generation sequencing technology bias and error on assembly.

Broader impacts: I propose to extend our existing efforts in interdisciplinary bioinformatics education to specifically address the underrepresentation of women and minorities in computational biology at the undergraduate level. This intertwining of education, outreach, and research is already inextricably part of my career, and a strong focus of the BEACON NSF Center here at MSU. We will develop a three-phase program of education, culminating in a co-mentored research experience in biology. Our long-term goal is to increase training, awareness, and participation in bioinformatics and computational biology among undergraduates taken from biology majors at MSU, North Carolina A&T, and elsewhere.

1 Background and Significance

1.1 Introduction and Overview

Shotgun sequencing uses random sequencing of DNA molecules to parallelize the process of sample preparation and is one of the most significant innovations in genome biology in the last 20 years, enabling the vast majority of genome projects. De novo assembly is part and parcel of shotgun sequencing: it takes sequence reads from shotgun sequencing and assembles them together into "contigs" based on sequence overlaps and mate-pair information, in the absence of a reference sequence. Because de novo assembly requires no reference sequence, it is the approach of choice when sequencing from new genomes. However, de novo assembly is computationally challenging, especially with large volumes of data, and is an extremely active area of research.

Biology plays a central role in genome sequencing. As sequencing becomes cheaper and easier, genomes from organisms with novel repeat structures, extreme GC/AT bias, and complex population structures are being sequenced. These genomic features challenge assemblers, which use heuristics that rely on sequence similarity and expect even sampling rates. Nonetheless great advances have been made in this area: sequencing and assembling a draft genome for a single organism can be achieved by single groups [Mortazavi et al., 2010].

As genome sequencing has become more straightforward, more complex samples are being sequenced. Both transcriptomes – the set of transcribed genes in one or more tissues – and metagenomes – mixtures of multiple different organisms – can be sequenced using the same shotgun approach as genomes. Often no reference sequence exists with which to analyze the sequencing reads. Hence de novo assembly approaches must be applied to make sense of the sequences; however, **transcriptomes and metagenomes possess a number of features that challenge genome assemblers, including high abundance variation in the source DNA or RNA population**. Eukaryotic transcriptomes may also possess many different isoforms from splice variants, which breaks the assumption of genome assemblers that many non-repetitive sequences will be used in only one final contig.

The challenge facing biology is how to effectively make use of the great advances in sequencing. We can now inexpensively sequence almost anything! This opens up entire new vistas in evolution, ecology, and medicine. Because much of what we sequence will never have been sequenced before, de novo assembly approaches are critical for working with this data. Below, I discuss the challenge of **scaling assembly approaches to these current and future data sets**.

In tandem with the technological challenge of making sense of sequence, we also face a social challenge: we lack the human capital to take advantage of these amazing opportunities in sequencing. So another challenge is how to grow educational infrastructure to train the next generations of biologists, who will of necessity be part *computational* biologist. This must be addressed with outreach and education.

1.1.1 Next-generation sequencing

Sequencing technologies are changing so fast that this paragraph will be seriously out of date by the time it is read! As I write, Roche 454 offers a million or more reads per run, of up to 600 bp average, in the $10k range. Illumina HiSeq machines can generate over a billion reads of 110-150 bp for the same cost. Newer technologies such as Ion Torrent and Pacific Biosciences offer cheaper and/or longer reads (e.g. PacBio promises 200,000 reads of length 1kb in the $200 range), but these are just starting to come into production [Schadt et al., 2010]. In this proposal I focus primarily on Illumina reads because Illumina sequencers are currently leading in depth of sampling.

These technologies enable biological investigations that would otherwise be difficult or impossible. Here, I focus on two applications: metagenomics and transcriptomics.

1.2 Metagenomics

Complex microbial populations participate in and drive many biochemical and geochemical processes, ranging from greenhouse gas flux, to nitrogen fixation in soil, to processing of gut nutrients, and beyond. Investigating the ecological and molecular principles underlying their participation in these processes is extremely challenging, because the vast majority of microbes exist in complex ensembles and cannot be cultured or studied in the lab. However, DNA and RNA from these assemblages can be readily extracted, and so one approach that has been developed over the last 15 years is to do gene-targeted or random (shotgun) sequencing of whole metagenomes [Tyson et al., 2004, Venter et al., 2004]. Random shotgun sequencing of metagenomes is known as “metagenomics”; **metagenomics is essentially the only way to completely characterize unculturable microbial assemblages** [Tringe and Rubin, 2005].

Sequencing technologies are only beginning to scale to the depth of sampling necessary to investigate metagenomic samples with a shotgun sequencing approach. With Sanger and Roche 454, low complexity communities can be investigated thoroughly and relatively cheaply [Tyson et al., 2004]. Medium complexity samples such as human gut and cow rumen can be sequenced to high coverage with Illumina today [Qin et al., 2010, Hess et al., 2011]. However, higher complexity communities such as soil and seawater possess thousands or millions of species of bacteria and archaea, and may require terabases of sequencing in order to fully sample low-abundance microbes. The complete number and extent of microbial communities is unknown, but one project proposes to generate petabases (10^{15}) or more of environmental sequence [Gilbert et al., 2010a].

One approach to metagenomic analysis focuses on targeted gene sequencing via PCR amplification from the community (“pyrotag” sequencing), which is biased towards genes we have already identified [Sogin et al., 2006]. Another approach analyzes raw sequencing reads using homology search to categorize likely gene content [von Mering et al., 2007]. This approach neither scales well to many reads, nor has high specificity with short reads. Eventually, single cell isolation and sequencing may provide a highly specific way to extract many genomes, but this is extremely challenging for low abundance microbes [Woyke et al., 2010].

De novo metagenome assembly approaches offer a number of advantages [Henry et al., 2011]. Assembly collapses short reads and produces contig sequences containing multiple genes and operons, facilitating computational analysis of putative protein function and ultimately synthetic biology approaches for investigation of metagenomic function [Llewellyn and Eisenberg, 2008, Gibson et al., 2008, Hess et al., 2011]. However, there are a number of associated challenges [Pignatelli and Moya, 2011].

1.2.1 Challenges in metagenome assembly.

Metagenome assembly faces two challenges in an era of large, short-read data sets. The first is *variable abundance of source organisms*, and the second is *scaling*. Variable abundance of the source organisms leads to variation in sequence sampling, which confuses assemblers that use “expected coverage” as a way to trim errors and detect repetitive sequence (e.g. Velvet’s approach, [Zerbino et al., 2009]. This is especially important with the increased depth of sampling necessary for assembly with short read sequencers, and potentially results in smaller and error prone assemblies due to misidentification of repeats. Some attempts have already been made to tackle this challenge using local coverage ([Peng et al., 2011], and MetaVelvet (unpublished)).

Considerably more work has been done on scaling, with two recent publications on de novo assembly of 200 GB+ short-read (Illumina) data sets. The first, on human intestinal tract bacteria (MetaHIT), sequenced samples from 100+ individuals [Qin et al., 2010]. The second sequenced cow rumen gut samples [Hess et al., 2011]. In both cases, very stringent abundance filtering was used to eliminate many reads and scale assembly, and a single set of overlap and coverage parameters was used. As no other approach yet exists for analyzing these data sets, it is difficult to estimate what effect these approaches had on the final assembly.

1.3 Transcriptomics

Good gene models and genome-wide gene catalogues are fundamental to understanding the development and physiology of organisms. Microarray and mRNAseq analysis of transcriptome expression both rely on good gene models [Pickrell et al., 2010, Li et al., 2010]. Without comprehensive gene models, even normalization for basic differential expression analysis is challenging. Unfortunately, assembling good gene models even for well-studied model organisms is challenging, because of the many physiological states and developmental tissues. For non-model organisms – the incredibly vast majority! – the challenges are even greater.

Sequencing technologies such as Illumina can now quickly and cheaply generate comprehensive transcriptome data sets from individual tissues, and even single cells [Tang et al., 2009]. Integrating multiple transcriptome data sets into a single reference set of gene models requires either a reference genome, or de novo transcriptome assembly. For reference-based approaches, the genome quality is critical for analysis; the genome must be largely complete and accurate, and contain relatively few differences from the organism being sequenced. This is true only for major model organisms, such as mouse, nematode, fly, arabidopsis, and yeast, where highly inbred lab strains are readily available. Even for important model organisms such as chick, zebrafish, and corn, the reference genome may contain errors, duplications, and misassemblies, all of which challenge reference-based approaches [Dodgson et al., 2011]. And the vast majority of biological organisms have neither a sequenced genome nor a substantial public transcriptome data set.

1.3.1 Current approaches in transcriptomic assembly

Several de novo assembly approaches are now available, including trans-ABySS, Oases, and Trinity [Robertson et al., 2010, Feldmeyer et al., 2011, Grabherr et al., 2011]. trans-ABySS extends the ABySS assembler to handle multiple assembly parameters, but does not address multiple isoforms within a single assembly run. Oases extends the Velvet assembler to output multiple isoforms based on expression variation and graph branching (unpublished; www.ebi.ac.uk/∼zerbino/oases/). Trinity focuses on an isoform partitioning approach to sensitively and specifically detect splice variants [Grabherr et al., 2011]. While all three produce superior results to reference-based approaches on at least some samples, none of the three specifically address scaling (ibid).

1.4 Assembly challenges: scaling and sensitivity

Scaling and sensitivity are the two most significant challenges for de novo assembly of both metagenomes and transcriptomes. In both cases, deep sequencing is required to detect low-abundance members of the population. This in turns requires that the assembly process handle substantial amounts of data; transcriptome sequencing also needs to be performed on many different samples, to detect tissue- or condition-specific genes.

Sensitivity is coupled to scaling because deep sampling is required to robustly sample rare transcripts or metagenome members. We have very little concrete idea of what increased sensitivity will bring us, because we are consistently pushing the boundaries by sequencing more and more deeply; it truly is tackling “unknown unknowns”. In metagenomic samples, we can detect 16s rRNA genes that appear to be from distinct species at an abundance of 1 in 100,000 or lower; these are impossible to assemble, yet these species may be ecologically significant members (e.g. “rare biosphere”, [Sogin et al., 2006]). The challenge for mRNAseq is similar: we know that many transcripts are expressed at low levels in small cell populations, and without physical enrichment or deep sequencing, we may not be able to detect them [Tang et al., 2009]. Many genes can provide significant function at low levels of expression, e.g. transcription factors at 10 transcripts/cell or fewer; these transcription factors could be critical regulators in developmental or physiological gene networks. **If genome sequence and transcriptome annotation is to be a foundation for hypothesis-driven biology, then sensitive detection of low-abundance variants is critical.**

Scaling is unfortunately quite difficult to achieve, for both theoretical and practical reasons. The primary block to scalability is memory: next-gen sequencing data sets generate enormous assembly graphs. The de Bruijn graph formalism relies on exact k-mer matches (fixed-length words of DNA) to implicitly detect overlaps, and hence scales with the number of unique k-mers [Pevzner et al., 2001, Miller et al., 2010]. Because they scale with sample novelty rather than with read number, De Bruijn graphs underlie the majority of recently developed assemblers, including ABySS, Velvet, SOAPdenovo, and ALLPATHS-LG [Miller et al., 2010]; but sufficiently deep transcriptome and metagenome sequencing overmatch these assemblers. For example, both the cow rumen and MetaHIT metagenome projects required $>$ 300 GB of RAM for less than 2bn reads, and Trinity requires an estimated 1 TB of RAM per 1 bn reads [Qin et al., 2010, Hess et al., 2011, Grabherr et al., 2011]. A number of unpublished and/or unproven algorithmic approaches to assembly scaling are now available but have unknown performance on metagenome and transcriptome samples [Simpson and Durbin, 2010, Ariyaratne and Sung, 2011].

The practical problem with assembly scaling lies in the pace at which new sequencing technologies are being developed. Sequencing technologies are now scaling faster than Moore’s Law, and it is possible to generate sequencing data sets much faster and more cheaply than it is possible to analyze or assemble them. Because of the depth needed for metagenomes and transcriptomes, computational hardware improvements are and will continue to fall behind the sequencing curve. Algorithmic advances in graph representation and traversal are needed for the future, rather than simply scaling hardware.

In the immediate future, Pacific Biosciences and other technologies will yield substantially longer reads with sufficient accuracy to be useful [Eid et al., 2009]. Unfortunately, **in addition to long reads, we need deep sampling for both metagenomics and transcriptomics**, to detect rare community members and transcripts. Longer reads will give us substantially *better quality* assemblies, but they are not a replacement for algorithmic advances in short read assembly. Probably the best future hope is that experimental advances in single cell sorting and sequencing will solve the sensitivity problem for both metagenomics and transcriptomics, but those technologies are not yet ready.

1.5 Significance and applications

De novo assembly approaches are a key method of analysis for many samples already being generated. Since Roche, Illumina, and Pacific Biosciences sequencers are within purchasing range of individual sequencing centers, and individual sequencing runs are in the $15k range, many academic and industrial research groups are generating their own genome, transcriptome, and metagenome samples; these groups often struggle to find computational resources and expertise capable of dealing with the sequence [Pennisi, 2011]. In addition, large scale sequencing programs like the JGI's Community Sequencing Program and the Earth Microbiome Project are generating vast amounts of metagenome sequence for environmental samples but provide relatively little in the way of computational support [Gilbert et al., 2010b]. This leads to a significant analysis gap between acquisition of sequence and any sort of analysis that can support hypothesis-driven research.

Scaling de novo assembly by an order of magnitude or more will be transformative. Scaling de novo assembly will enable the use of rental or "cloud" computers (e.g. Amazon Web Services), thus democratizing assembly for small groups [Schatz et al., 2010]. It will also permit more and faster iterations with different parameter sets, improving assemblies. And it will leverage CPU power, enabling more and better graph reduction heuristics to be applied.

The applications of better de novo assembly approaches in both metagenomics and transcriptomics are hard to overstate. Deeper, faster, and better metagenomics and transcriptomics will applicable to nearly every facet of evolutionary research, ecological research, medical research, and environmental processes. In particular, more and better reference genomes for metagenomics, and more and better reference transcriptomes for plants and animals, would enable perturbation studies at an environmental and ecological level, provide for better evolutionary analysis of developmental processes, and improve agricultural and medical investigations.

2 Research proposal

Preliminary results: development of a lightweight, compressible de Bruijn graph representation

Driven by the need to scale metagenomic sequence assembly to commodity hardware, we have developed a lightweight probabilistic de Bruijn graph representation. This de Bruijn graph stores k-mer nodes in Bloom filters and keeps edges between nodes implicitly, i.e. if two k-mer nodes exist with a k-1 overlap, then we say that there is an edge between them. Because Bloom filters are a probabilistic set storage data structure, with false positives (but no false negatives), there is another false positive rate associated with the false presence of nodes [Knuth, 1975]. This graph structure is effectively **compressible** because one can choose a larger or smaller size for the underlying Bloom filters; a larger size admits fewer node false positives, while a smaller size admits more.

This compressible de Bruijn graph representation possesses a number of useful properties. By relying on Bloom filters, it is **constant memory**: no additional memory is used as additional data is added. Another interesting property is that as the graph structure degrades (due to decreased memory or increased data), only false positive nodes and edges are gained, so compressing the graph results in a more tightly interconnected graph; the intuition is similar to that of crumpling a piece of paper more and more tightly.

The most intriguing and useful result, however, is that while the *local* structure of the graph degrades linearly with compression, the *global* structure of the graph is extremely accurate up until a

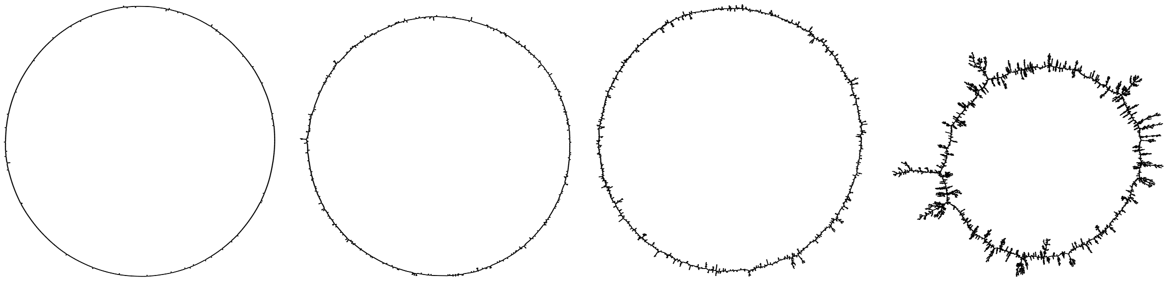

Figure 1: Graph compression effects on global structure of a circular genome with (from left to right) false positive rates of 1%, 5%, 10%, and 15%. Local structure elaborates but global structure remains the same.

certain point, when it catastrophically collapses. This catastrophic collapse qualitatively and quantitatively resembles lattice percolation, in which at a certain node probability p an infinitely sized lattice is completely connected; equivalently, the graph undergoes a first order phase transition in connectivity. For de Bruijn graphs, which are finite, the graph ends up being dominated by a single, highly connected lump. The false positive rate at which this collapse occurs is easy to calculate and can be explained intuitively as the point at which each individual node is likely to have at least one false edge, $f_p \approx \frac{1}{8}$. Measures of global graph connectivity - in particular, the shortest longest path - are constant until close to this percolation threshold, indicating that the large scale graph structure is faithful; see Figure 1.

This compressible graph representation can thus faithfully represent long-range connectivity in de Bruijn graphs with a very small memory footprint of between .5 - 2 bits per k-mer, or 1-4 GB per human genome. In particular, it *can* be used as a lightweight representation for exploring graph properties and analyzing extremely large data sets.

Preliminary results: connectivity analysis of metagenomic and mRNAseq data sets

Local graph density G at distance N is a measure of graph connectivity; a de Bruin graph region built from a linear sequence has a graph density of 2, with branches or repeats increasing the number. We compared the local graph densities of several large metagenomic data sets to the graph density of all 600 microbial genomes from NCBI using k=32. At a distance N of 100, fewer than 6% of the nodes in the microbial genomes graph had an average graph density greater than 20, while more than 44% of the nodes in the metagenomic samples did; see Figure 2. In a similar comparison, over 40% of the nodes in an mRNAseq read set had G $>$ 20, while the chicken RefSeq transcriptome had fewer than 5%. **These results indicate the presence of substantial spurious connectivity in both our metagenomic and mRNAseq data sets.**

We hypothesized that this spurious connectivity was due to systematic biases in base calling, and should therefore be non-uniform with respect to read position. As shown in Figure 2, for metagenomic data sets, average graph density for N=10 rises with the position of the starting k-mer in the read, strongly suggesting that it is due to base calling bias. However, for mRNAseq data, average graph density is raised by simple sequence repeats, perhaps introduced during sample preparation.

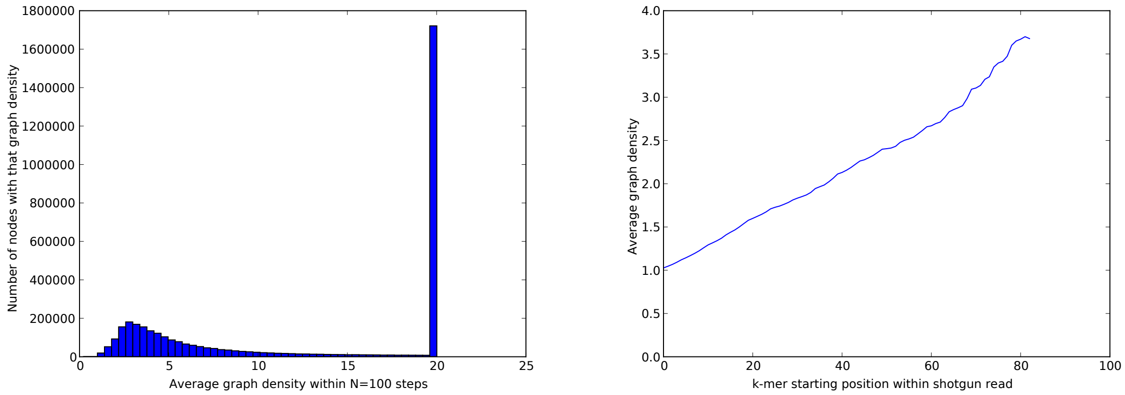

Figure 2: Left: a histogram of average graph density G within 100 steps of the center of every read in a 4m read data set, with $G > 20$ aggregated at $G = 20$. Right: graph density within 10 steps of every k-mer, by position within shotgun read.

Preliminary results: removal of highly connected k-mers

Suspecting that these artificial repeats were potentially causing assembly problems, we implemented a systematic traversal algorithm to identify highly connected k-mers, that is, k-mers that are reachable from many locations in the graph. Briefly, we built waypoints that cover the graph at a distance $L \approx 40$, and then systematically traversed all k-mers within L of each waypoint. Excursions that covered more than 200 k-mers (for a graph density G of $200/40 = 5$) were labeled as ”big excursions”; any k-mers that show up in more than 5 big excursions were labelled as “bad”, and reads containing them were truncated. These k-mers show a significant 3’ bias in their position in reads and are hence likely to be artifacts; the resulting density graph looked much more similar to assembled genomes.

This novel lightweight graph representation is a computational advance that readily enables the following aims.

Aim 1: Improve and scale metagenomic sequence assembly

Preliminary results: scaling metagenomic assembly with partitioning

Metagenomic data sets originate from samples with many largely disconnected DNA sequences – individual microbial genomes. We can exploit this biological structure by partitioning the assembly graph into disconnected subgraphs that represent the original DNA sequence components. This scales the assembly problem by reducing the bulk assembly to many individual smaller assemblies. Moreover, for de Bruijn graphs, partitioning can be applied iteratively: we can partition once at small k_p (say, 20) and assemble at any larger k, and still get equivalent assemblies. Finally, this partitioning approach has another advantage, which is that we can optimize assembly parameters for individual partitions or source contigs. While partitioning approaches have already been used in metagenome assembly, these approaches use a standard de Bruijn graph ([Peng et al., 2011]; also see MetaVelvet (unpublished)). **These previous approaches do not scale: the memory usage of the entire assembly process is dominated by the need to load the initial graph into memory.**

We therefore implemented graph partitioning using the compressible graph representation described above. Briefly, we use the input sequences from which the graph originates as a guide to mark waypoints in the graph at a minimum density, and then use breadth-first search to exhaustively connect the waypoints. Each set of connected waypoints forms a single graph partition. A second pass through the input sequences then places the sequences into partition bins, which can be assembled individually. Because the compressible graph representation admits only false positive nodes and edges, no sequences will be incorrectly detached from their true partition.

We first applied this to a small test data set of 35m 110x2 Illumina reads from a soil sample from an Iowa corn field. We partitioned with k=32 in 2gb of RAM, resulting in 2m reads with some connection to at least one other read. The largest partition contained 330k reads. **Individual assembly of the partitions in this much-reduced data set with ABySS yielded identical results to assembly of the entire 35m read data, demonstrating the effectiveness of partitioning.** The overall assembly time was approximately the same, and partitioning+assembly required only 2gb of RAM, while assembly of the entire data set required approximately 40gb of RAM, for a 20x scaling factor.

We next applied partitioning to a 50 Gb soil sample from an Iowa corn field, provided by the Great Prairie Grand Challenge project (see letter of collaboration). This sample consists of over 650m 110x2 reads from Illumina GAII, sequenced by JGI from a 2cm soil plug taken at a depth of 5 cm. We estimate that this sample contains approximately 3×10^{10} unique 32-kmers. We filtered, partitioned, and assembled this sample at k=32 in 68gb of RAM on an 8-core Amazon rental computer in 60 hours. After eliminating partitions with fewer than 5 reads, we were left with approximately 250m reads in 200k partitions, with the largest partition containing approximately 100k reads. These partitions collectively assembled into 354k contigs $>$ 500bp, containing 277 Mb altogether. A bulk (unfiltered, unpartitioned) assembly of the same sample done with the same parameters on our local HPC required more than 270gb of RAM (a 4x increase) and took approximately the same amount of time.

Comparing our two assemblies, we were surprised to find that assemblies built with the same parameters shared less than 50% of constituent k-mers in contigs $>$ 500 bp; that is, **filtered and unfiltered assemblies were less than 50% similar**. Since the only lossy operation we had conducted was artifact filtering, we theorized that the bulk assembly was likely to contain missassemblies around these artifacts. In support of this, we found that open reading frames (ORFs) from the bulk assembly tended to end at artifacts (5% over background). We also analyzed less diverse/higher coverage cow rumen and MetaHit short read metagenomic data sets; these data sets contained artifacts at similar frequency to the soil data set. However, when we filtered, partitioned, and assembled subsets of both rumen and MetaHit, we found that the bulk assemblies were identical in composition to the filtered/partitioned assemblies. Thus assembly of high coverage data sets appears to be refractory to these artifacts. Our overall model is that **overlapping artifacts in sequences from our highly diverse soil sample (with its extremely low coverage) are competing with genuine overlaps for assembly, and are thus causing misassemblies**. In high-diversity samples, these artifacts can nucleate the linkage of two short contigs, resulting in dramatically different bulk assemblies from the filtered assemblies. However, in lower diversity/higher coverage samples these artifacts do not appear at sufficient frequency to confuse the assembler. Other samples and assemblers present with the same issues.

Preliminary results: improving metagenomic assembly with per-partition optimization

Noting that per-partition coverage for the Iowa corn sample varied widely (between 2x and 10x k-mer coverage), we hypothesized that we could improve the Iowa corn assembly by choosing parameters on a per-partition basis. We therefore used an exhaustive search strategy across multiple k values to assemble each partition optimally, with the objective of maximizing the sum(contigs $>$ 1kb). We found that optimal values for k ranged from 21-51, and correlated with k-mer coverage. This resulted in a more than 10-fold increase in assembled contigs from low-coverage partitions, and a 20% increase in the average contig size for high-coverage partitions. **These results strongly reinforce the idea that current assemblers do not perform optimally on bulk metagenomic data sets because of the internal variation in coverage.**

Based on these preliminary results, we propose to develop and apply our partitioning approach to scale metagenomic assembly, with the following three subaims:

Subaim 1.1: Apply partitioning to many metagenomic samples

We will adapt and apply our filtering and partitioning approach to existing and emerging short-read data sets from a variety of samples, with the twin goals of **generating good assemblies from large, challenging data sets of wide biological interest** and **exploring software parameters for our filtering approach**. We know of no demonstrated approaches for lossless scaling of metagenome assembly, which makes this a great target of opportunity. In collaboration with the Earth Microbiome Project and others we will analyze a variety of samples from a spectrum of environments and biological sources. We will start by reanalyzing well-analyzed samples (rumen and MetaHIT) and scaling our approach to handle larger amounts of soil data; in all of these cases we now have well over 200 Gb of Illumina short-read data, which challenges our filtering and partitioning implementation. Post-assembly analysis of the samples will be performed by our collaborating labs; annotation of assemblies can easily be done in MG-RAST v3, IMG/M, or CAMERA [Glass et al., 2010, Markowitz et al., 2008, Sun et al., 2011].

Subaim 1.2: Apply partitioning to other sequencer data

We will also extend our software and graph analysis approaches to work with existing and emerging types of sequencing data. **Given the diversity of sequencing technologies already on-line and the promise of many more in the near future, integration of data from different technologies is a critically important endeavor for metagenomics.** Most existing Sanger and 454 metagenomic data sets are quite shallow, but integration with Illumina may help assemble difficult-to-sequence regions. Looking to the future, 454, Pacific Biosciences (PacBio) and Ion Torrent are all promising longer reads at substantially lower costs. These technologies do not yet promise *deep* sampling of high diversity samples like soil, but should be helpful in assembly. However, their biases and error structures are not yet well understood, and this is where the challenge will lie for partitioning. We will integrate 454 and PacBio data into our Illumina partitions and coassemble using Velvet; this will require new heuristics and filtering parameters.

Subaim 1.3: benchmark assembly approaches

We will develop benchmark data sets from our partitioning approach and evaluate other existing assemblers on them, including SOAPdenovo, ALLPATHS-LG, SGA, and PE-Assembler [Peng et al., 2011]. **Understanding what technologies work "best" for metagenomic sequences,**

evaluating them openly, and developing openly available benchmark data sets, are approaches that are critical to advancing the field (also see assemblathon.org). We will define reference sets; build tools to compare assemblies; and make the data sets, tools, and our own assemblies available via Amazon EC2/S3/EBS. (See Data Management Plan.) This will enable other computational groups to run our entire data set through our tools, as well as enabling their own assembly and comparison approaches.

Aim 2: Scale and improve mRNAseq assembly

mRNAseq is a separate and substantial set of challenge from metagenomics, but there is some similarity in the challenges to be faced. We therefore propose to modify the tools and techniques we develop for metagenomics and apply them to mRNAseq data sets.

Preliminary results: scaling mRNAseq assembly

mRNAseq, like metagenomic data sets, contains non-biological repeat structures that lead to very dense graph structures (Preliminary Results). However, unlike metagenomic data, these repeat structures do not have a 3' bias in the sequencing reads and hence may not result from base calling bias. Nor did the filtering parameters developed for metagenomic data sets remove the repeat structures completely. **However, repeat filtering *did* succeed in scaling mRNAseq assembly dramatically.**

We assembled 3 lanes of 76x2 mRNAseq sequenced by an Illumina GAII machine (totaling 120m reads) from embryonic stages of an ascidian, *Molgula oculata*; the RNA was extracted in our lab. Assemblies were performed on unfiltered and filtered data, using Velvet and Oases [Zerbino and Birney, 2008]. While the filtering removed over 30% of the reads, the assemblies of filtered and unfiltered contained essentially the same gene set based on a reciprocal best-hit BLAST analysis. More impressively, the assembly of the filtered reads required only 40gb of RAM and took only 5 hours (3x less memory and 5x less time than the assembly of the unfiltered reads). Similar results have been achieved with mRNAseq from *G. gallus* (chicken), *P. marinus* (lamprey), and a number of nematode data sets (see letter). **Graph density filtering generally scales mRNAseq assembly 5-20x.**

We examined the highly connected k-mers identified by our traversal approach and found that they contained a substantial "cloud" of poly-A like sequences, as well as other simple sequence tracts. These probably stem from legitimate sequence (such as poly-adenylated mRNA sequences) as well as PCR artifacts engendered by the RT and PCR amplification stages of the standard mRNAseq protocol from Illumina. While these may be useful in post-assembly steps (e.g. identification of poly-adenylation sites [Pickrell et al., 2010]), they appear to engender repeat structures that challenge Velvet and Oases. We see these sequences in data sets from multiple sequencing centers, suggesting that they may be a general presence in mRNAseq sets. They are generally not present in simulated data sets (e.g. [Jackson et al., 2009]).

Preliminary results: local assembly approaches improves isoform and gene detection

We performed several de novo assemblies of chicken mRNAseq from Illumina (56x1) using Velvet and Oases. As reported by other groups we found that de novo assembly identified new genes and isoforms that referenced-based approaches missed [Robertson et al., 2010, Grabherr et al., 2011, Woyke et al., 2010].

Observing that global assembly was slow enough to challenge our ability to run multiple different assemblies per sample for different k, we developed a local assembly approach based on mapping the reads to genomic loci and then assembling reads by chromosome. While this did indeed scale assembly, it also resulted in the appearance of a number of novel genes and isoforms, most of reasonably low abundance. These isoforms were "hidden" by repetitive artifacts not present in reads that map to the genome; density analysis confirms that the graph connectivity of mapped reads is significantly lower. These isoforms were confirmed by inspection of reads, as well as with comparison with chicken ESTs from 454 and Sanger. BLASTing chrI genes against the mouse RefSeq database (cutoff 1e-20) showed a 15% increase in gene predictions using the local assembly approach (650 genes from global, + 115 extra from local).

Based on these preliminary results, we propose three subaims for scalable and sensitive mRNAseq assembly:

Subaim 2.1: Improve mRNAseq repeat filtering

We will develop reference-free repeat identification and trimming approaches for mRNAseq, to aid in de novo assembly of mRNAseq data sets. We expect this to result in dramatically "cheaper" assemblies that are significantly more sensitive to low-expressed genes and isoforms. Specifically, we will adapt our filtering software to use parameters that trim reads based on graph properties of the sequence. Results will be evaluated as above.

Subaim 2.2: Develop reference-free and locus-specific partitioning

We will develop a pre-assembly partitioning approach for mRNAseq that does not rely on a pre-existing assembly, unlike existing mRNAseq assemblers [Grabherr et al., 2011]. Specifically, we will break the de Bruijn graph at repetitive k-mers while using read threading to connect partitions. Once we partition reads on a per-locus basis, we can assemble the partitions individually. **This will allow us to assemble essentially arbitrary amounts of mRNAseq sequence.**

Subaim 2.3: Use graph-based merging to remove redundancy

The many redundant sequences resulting from assembling with multiple k values are difficult to collapse without a reference, but should be amenable to merging with a de Bruijn graph. We will develop a de Bruijn graph approach to take overlapping isoforms and do a path-based analysis that merges exact or near-exact prefixes and suffixes. For validation, it will be possible to identify likely exon boundaries based on diverging paths in the graph, and compare these against reference sequence for chicken and lamprey samples.

Aim 3: Subdividing large de Bruijn graphs for piecemeal assembly

The connectivity filtering and partitioning approach proposed in Aims 1 and 2 are specific to metagenomic and mRNAseq data sets. In particular, lossless partitioning is made possible by the *biological* structure of the data. These approaches deal with a pressing need to improve the analysis of extant data sets. However, more general approaches to graph partitioning that do not depend on the specific nature of the data could be very helpful in assembly research.

Assembly has always been a frustratingly irreducible problem, requiring that large graph structures be created from massive data sets before any graph compression or partitioning heuristics can be applied. While many implementations distribute assembly or assembly graphs across multiple nodes they generally either do not use graph locality to do so, or else must use local heuristics to infer graph locality prior to distribution or partitioning, which is itself a challenging problem [Kingsford et al., 2010, Emrich et al., 2004, Simpson et al., 2009] (and see also Contrail (M. Schatz, unpublished; contrail.sf.net)).

Our lightweight graph representation allows us to easily load and analyze the connectivity structure of extremely large graphs. This connectivity structure can then be explored to detect easily compressed or partitioned portions of the graph, as well as provide physical locality of k-mer nodes that are close in the graph. We therefore propose:

Subaim 3.1: Develop modularity measures for extremely large graphs

We will use our lightweight de Bruijn graph construct to analyze the modularity of assembly graphs and look for minimally interconnected modules. Our initial approach will use a greedy local algorithm to identify and exhaustively explore regions of the graph with low external connectivity, and combine that with a repeat "tangle" identifier to identify likely repetitive regions [Do et al., 2008, Novak et al., 2010, Gu et al., 2008]. The results from Aim 1 and 2 will provide example data sets and parameters for initial exploration.

Since many existing graph assemblers already use local heuristic approaches to iteratively transform the graph into modular structures (e.g. Velvet [Zerbino et al., 2009] and [Kingsford et al., 2010]), this is a robust approach. The novelty in our approach is in the use of our graph representation, which allows us to scale the process.

Subaim 3.2: Implement a framework for extracting, assembling, and reintegrating independently assembled modules.

We will extract reads belonging to minimally interconnected modules for independent assembly. Connectivity information retained during the extraction can be provided to an independent scaffolder such as Bambus2, for reintegration of piecemeal assemblies into a global assembly [Treangen et al., 2011] (also see Bambus2 (unpublished) at www.cbcb.umd.edu/software/bambus/). One significant advantage of this approach is that different assemblers can be used to assemble different local portions of the graph. So, for example, regions with high heterozygosity could be fed into an overlap assembler, while specific repetitive regions could be deconvolved with a guided approach like EMIRGE [Miller et al., 2011]. Long-insert paired ends can be individually used in assembly and then used again in the scaffolding step. We propose to integrate this approach into the AMOS assembly pipeline (see letter of collaboration from Mihai Pop).

A general, scalable approach to annotating structures in large assembly graphs would have significant implications for assembly scaling and quality. In particular, being able to distribute graph components across smaller memory machines could enable substantially more compute intensive heuristics for complex graph resolution.

Subaim 3.3: Develop approaches for iterative assembly.

Analyzing modularity in assembly graphs, and breaking large graphs down into minimally connected graph components, also enables more effective iterative assembly approaches. **Given the current ease of resequencing and the rapid development of new sequencing technologies, approaches for improving assemblies, comparing assembled regions, and analyzing resequencing data sets are going to be desperately needed.** Specifically, we propose to compare modules in existing assemblies to modules extracted from new data sets. Modules that would extend the existing assembly, or otherwise add new information, can be used to (re)assemble the relevant region. Modules that are consonant with the existing assembly can simply be discarded.

This also allows for the easy comparison of assemblies against new sequencing data and new sequencing data types.

Additional work: Implementation improvement

There is substantial room for improvement of our basic graph analysis software, khmer, which is written in C++ and Python. The khmer software is already freely available on github.com, contains 130 unit tests and several tutorials, and is being used by about a dozen labs; but it is still relatively immature. In particular, there are several avenues for optimization of the core data structure; the unit tests cover the core functionality but not many of the associated scripts; and the existing tutorials do not adequately cover the internal structure of khmer, the choice of parameters for analyses, and the output diagnostic formats. Optimization of core data structures should result in a 2-10x speed improvement for traversal, and rewriting the ancillary data structures should result in a 50-80% decrease in memory usage for partitioning.

3 Additional Information

Previous Support

RUI:MSB:Collaborative: Symbiont separation and investigation ... This is a collaborative NSF grant ($180k total for two years, ending 12/31/2011) with Dr. Goffredi at Occidental College, sequencing microbial symbionts from *Osedax*, a marine polychaete. Preliminary results for this grant include a phylogenomic evaluation of homology methods for short reads, and the resulting false positive rate (J. Guo and C.T. Brown, in preparation). As the primary sequence data from this grant has yet to be generated, we will be requesting a no-cost extension for the 3rd year of the grant.

Personnel/Work summary and breakdown

Dr. C. Titus Brown (the PI) will supervise all software development, research, and outreach aspects of this grant.

We also request support for one graduate RA from Computer Science, who will lead the theoretical and computational aspects of the research as well as coordinate the outreach effort. For years 1-2 this will be **Jason Pell**, who has done most of the graph theory and percolation analysis for our data structure. For years 3-5, we will recruit a new graduate student, perhaps Meghan Donahue, a Computer Science undergraduate from Indiana University who contributed to Specific Aim 2/Preliminary Results during her summer research.

In addition to Mr. Pell, a number of additional people have done and will do work on assembly with no additional cost to this grant. **Dr. Arend Hintze** is a 5th-year postdoctoral fellow in the Brown Lab, partially funded by a USDA grant to build Web interfaces for next-gen sequence analysis, and also funded by the Institute for Cyber-Enabled Research and BEACON. Dr. Hintze has helped drive the graph theory and percolation theory research together with Mr. Pell. **Dr. Adina Howe** is a 2nd-year postdoctoral fellow in the Brown Lab, shared with the Tiedje lab at MSU. She is funded by an NSF postdoctoral fellowship, and will continue to be funded by the Great Lakes Bioenergy Research Center to do metagenome assembly work. She developed much of the preliminary results for Aim 1, Metagenome Assembly. **Rosangela Canino-Koning** is a

2nd-year CSE graduate student with funding from an NSF graduate fellowship. She is completing her Masters on probabilistic data structures in assembly this year. **Likit Preeyanon** is a 3rd-year Microbiology graduate student funded by a Thai government fellowship. He has developed reference-based assembly approaches for mRNAseq, and developed the preliminary results for Specific Aim 2, mRNAseq Assembly. **Elijah Lowe** is a 3rd-year CSE graduate student funded by the Gates Foundation and the BEACON Center to do experimental work on chicken and ascidian development. He will work on biological validation of predicted gene isoforms for both ascidian and chick mRNA.

Timeline

Year 1: metagenome partitioning and long-read integration.
Year 2: metagenome benchmark development; mRNAseq repeat filtering and partitioning.
Year 3: final metagenome work; mRNAseq isoform collapse; begin of modularity analysis.
Year 4: module analysis, extraction, and assembly.
Year 5: integration with Bambus2/AMOS; iterative assembly.

4 Outreach plan

I propose to target the bioinformatics gap that exists between computer science and molecular biology/genomics by building an undergraduate summer research program that combines education with research in biology, genomics, and bioinformatics. There are two gaps to be closed. First, there is a substantial underrepresentation of women and minorities in bioinformatics. This is, at heart, a “supply” problem: the dearth of women and minorities in computer science undergraduate studies leads to an underrepresentation in CS-derived courses of study. Second, there is a lack of students conversant in both CS and biology.

These gaps introduce an opportunity for training undergraduates in practical research issues. There are many basic computational research tasks that are readily accessible with little training and some hands-on guidance. These research tasks include assembling and annotating microbial genomes, BLASTing large sequence collections for matches to given proteins, and mapping short reads to existing genomes for resequencing analysis. With appropriate mentoring, these tasks provide an easy opportunity for undergraduates to engage in real research.

I propose to use my unique position as a bridge between the CSE and Microbiology departments, and my role as an educator in the BEACON NSF Science and Technology Center on the Study of Evolution in Action, to address these gaps. My teaching and education activities at MSU have already focused on this at both the undergraduate and graduate level. I was hired to help bridge the biology and computational programs here, and have already initiated a number of courses at the graduate level or above. I developed an interdisciplinary graduate seminar course in bioinformatics that achieved an approximately 50/50 split between 25 students from bio and CS majors in its second year; I also developed ”Computational Science for Evolutionary Biologists”, a grad course taught across three of the BEACON campuses (two via teleconference, UT Austin and UW Seattle) last year and across four campuses (U. Idaho) this next year. I am also the founding course director for a summer workshop for career biologists on Analyzing Next-Generation Sequencing Data that attracted over 130 applicants (including 15 faculty, 50 postdocs, and 50 graduate students) in 2011. The final student body of 24 included 17 women. In all of these educational situations I have developed approaches to gently introduce biologists to computation, and have done so with

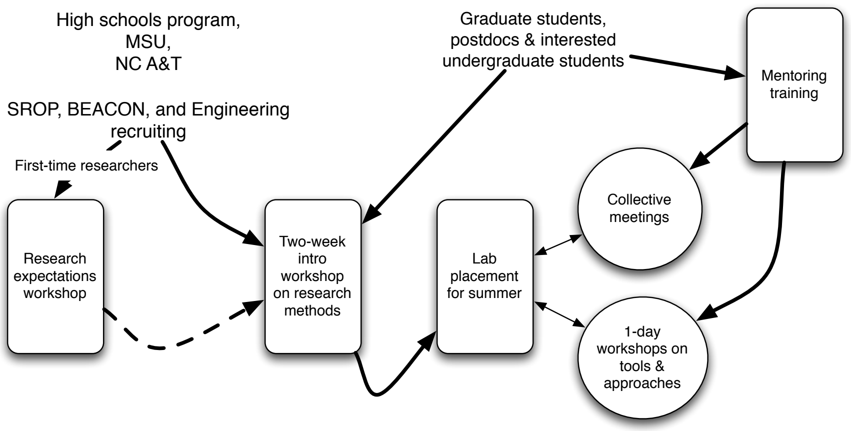
High schools program,
MSU,
NC A&T
SROP, BEACON, and Engineering recruiting
First-time researchers
Graduate students, postdocs & interested undergraduate students
Mentoring training
Collective meetings
Research expectations workshop
Two-week intro workshop on research methods
Lab placement for summer
1-day workshops on tools & approaches

Figure 3: Proposed outreach plan.

consistently high student evaluations.

I will develop a three-tier program in mentoring. First, I will create a summer program that recruits sophomore women from the Genomics and Molecular Genetics undergraduate major at MSU and sophomore students from NC A&T. This summer program will start with a two-week workshop on basic computational research skills. Second, I will place students in biology labs with computational projects. During this they will be co-mentored by members of my lab. Third, in a summer series of 1-day workshops, we will delve further into bioinformatics. In doing so we hope to interest them in additional computational or mathematical education in their coursework, e.g. a CS minor. Finally, graduate students and postdocs from my lab – many of whom have transitioned between disciplines themselves – will be given formal mentoring training and will engage in mentoring and teaching activities, providing role models for the undergraduates and gaining valuable career experience.

By giving women and other underrepresented minorities a strongly guided research experience in bioinformatics the summer before they become juniors, we will position them well to capitalize on their experience through the next summer, when they will be exploring career options in research or industry. Moreover, in year 2, we expect to invite interested and successful "graduates" from the first summer program to participate in both the education and research programs in the second summer. Note that one potential source of students is a high school training effort that just started this past summer; the BEACON program is developing follow-on programs and has asked me to help coordinate them.

Implementation: The BEACON NSF STC and the Summer Research Opportunities Program (SROP) will provide administrative support and help coordinate administration and funding; see letter of collaboration. Several of my graduate students, including Jason Pell and Elijah Lowe, and my postdoc Adina Howe, will coordinate and teach the two-week workshop, based on exercises derived from my existing fall course. Once the summer undergraduate students are placed in research labs, we will hold weekly meetings collectively to discuss progress and tackle problems as a group. We will also conduct a number of 1-day intensive compute workshops to tackle specific analysis issues. These workshops will be adapted from source materials already available from our two-week workshop, adapted to provide more scientific background.

Data Management Plan

We expect to generate 3 types of data and source code: various assemblies and associated parameters; assembly evaluation pipeline code; and khmer source code.

Assemblies and associated parameters will be archived on Amazon S3 (Web URLs) and/or Amazon EBS (virtual hard drives/snapshots) and made publicly available at the time of submission of papers. Funding is requested to support data archiving on Amazon at $.10/gb/mo (see Budget). We have extensive experience with Amazon Web Services. Past the period of grant support, we will ask Amazon to host our data sets as part of their public data set archive.

Assembly evaluation pipeline code and software source code will be checked into a public git repository at github.com/ctb/. Results (assemblies, etc.) will reference the specific version within the git repository. We will also periodically create public Amazon Machine Images (AMIs) that come preinstalled with specific software releases and can be run on Amazon's Elastic Cloud Computing resource.

No primary sequence data will be generated from this grant, but we will make data used for assemblies public along with the assemblies. We will publish assembled data sets through Standards in Genomic Sciences (SIGS), for citation handles and methods archiving. As we are working with people from several communities, we will also submit transcriptome and metagenome assemblies to community standard archives as appropriate; specifically,

- Our soil metagenome assemblies from the Great Prairie Grand Challenge project will be placed in MG-RAST v3 and made available through their interface, as will all EMP-derived assemblies.
- We are co-authors in the lamprey genome annotation effort and will provide an official transcriptome for it through NCBI.
- The ascidian transcriptome will be provided to CIPRO, the Ciona protein database.
- The chicken transcriptome reannotation will be provided in UCSC Genome Browser compatible format (BED and WIG files) on an Amazon S3 site, so that they can be used directly as an annotation track on the UCSC Genome Browser.

Assemblies and metagenomes from the MetaHIT and Human Microbiome Project may be subject to additional disclosure rules because of HIPAA and European privacy regulations. Those will be dealt with according to the MetaHIT and HMP data release rules as they are developed.

Facilities, Equipment, and Other Resources

Facilities

Dr. Brown has both computational and wet lab space.

The wet lab space consists of 1000 sq ft with a hood, benches for 6 students, and several freezers. The lab is equipped for chick embryology work with a fluorescent dissecting scope, two egg benches, and two incubators. The lab is also fully equipped with microcentrifuges, PCR machines, water baths, and a microwave.

The computational space consists of 2500 sq ft shared with another computational biology lab. There are offices for 10 students and postdocs, two faculty, and a large central collaboration space.

Equipment

The lab has three Dell/Intel servers, each with 16 GB of RAM, on which primary development work and testing is done. We also have access to the High Performance Compute facility at MSU, which possesses over 4,000 CPUs in chassis with up to 24 GB of RAM, and several high memory computers available on a scheduled basis.

References

[Ariyaratne and Sung, 2011] Ariyaratne, P. and Sung, W. (2011). PE-Assembler: de novo assembler using short paired-end reads. Bioinformatics *27*, 167–74.

[Do et al., 2008] Do, H., Choi, K., Preparata, F., Sung, W. and Zhang, L. (2008). Spectrum-based de novo repeat detection in genomic sequences. J Comput Biol *15*, 469–87.

[Dodgson et al., 2011] Dodgson, J., Delany, M. and Cheng, H. (2011). Poultry genome sequences: progress and outstanding challenges. Cytogenet Genome Res *134*, 19–26.

[Eid et al., 2009] Eid, J., Fehr, A., Gray, J., Luong, K., Lyle, J., Otto, G., Peluso, P., Rank, D., Baybayan, P., Bettman, B., Bibillo, A., Bjornson, K., Chaudhuri, B., Christians, F., Cicero, R., Clark, S., Dalal, R., Dewinter, A., Dixon, J., Foquet, M., Gaertner, A., Hardenbol, P., Heiner, C., Hester, K., Holden, D., Kearns, G., Kong, X., Kuse, R., Lacroix, Y., Lin, S., Lundquist, P., Ma, C., Marks, P., Maxham, M., Murphy, D., Park, I., Pham, T., Phillips, M., Roy, J., Sebra, R., Shen, G., Sorenson, J., Tomaney, A., Travers, K., Trulson, M., Vieceli, J., Wegener, J., Wu, D., Yang, A., Zaccarin, D., Zhao, P., Zhong, F., Korlach, J. and Turner, S. (2009). Real-time DNA sequencing from single polymerase molecules. Science *323*, 133–8.

[Emrich et al., 2004] Emrich, S., Aluru, S., Fu, Y., Wen, T., Narayanan, M., Guo, L., Ashlock, D. and Schnable, P. (2004). A strategy for assembling the maize (Zea mays L.) genome. Bioinformatics *20*, 140–7.

[Feldmeyer et al., 2011] Feldmeyer, B., Wheat, C., Krezdorn, N., Rotter, B. and Pfenninger, M. (2011). Short read Illumina data for the de novo assembly of a non-model snail species transcriptome (Radix balthica, Basommatophora, Pulmonata), and a comparison of assembler performance. BMC Genomics *12*, 317.

[Gibson et al., 2008] Gibson, D., Benders, G., Axelrod, K., Zaveri, J., Algire, M., Moodie, M., Montague, M., Venter, J., Smith, H. and 3rd Hutchison CA (2008). One-step assembly in yeast of 25 overlapping DNA fragments to form a complete synthetic Mycoplasma genitalium genome. Proc Natl Acad Sci U S A *105*, 20404–9.

[Gilbert et al., 2010a] Gilbert, J., Meyer, F., Antonopoulos, D., Balaji, P., Brown, C., Brown, C., Desai, N., Eisen, J., Evers, D., Field, D., Feng, W., Huson, D., Jansson, J., Knight, R., Knight, J., Kolker, E., Konstantindis, K., Kostka, J., Kyrpides, N., Mackelprang, R., McHardy, A., Quince, C., Raes, J., Sczyrba, A., Shade, A. and Stevens, R. (2010a). Meeting report: the terabase metagenomics workshop and the vision of an Earth microbiome project. Stand Genomic Sci *3*, 243–8.

[Gilbert et al., 2010b] Gilbert, J., Meyer, F., Jansson, J., Gordon, J., Pace, N., Tiedje, J., Ley, R., Fierer, N., Field, D., Kyrpides, N., Glockner, F., Klenk, H., Wommack, K., Glass, E., Docherty, K., Gallery, R., Stevens, R. and Knight, R. (2010b). The Earth Microbiome Project: Meeting report of the '1 EMP meeting on sample selection and acquisition' at Argonne National Laboratory October 6 2010. Stand Genomic Sci *3*, 249–53.

[Glass et al., 2010] Glass, E., Wilkening, J., Wilke, A., Antonopoulos, D. and Meyer, F. (2010). Using the metagenomics RAST server (MG-RAST) for analyzing shotgun metagenomes. Cold Spring Harb Protoc *2010*, pdb.prot5368.

[Gnerre et al., 2011] Gnerre, S., Maccallum, I., Przybylski, D., Ribeiro, F., Burton, J., Walker, B., Sharpe, T., Hall, G., Shea, T., Sykes, S., Berlin, A., Aird, D., Costello, M., Daza, R., Williams, L., Nicol, R., Gnirke, A., Nusbaum, C., Lander, E. and Jaffe, D. (2011). High-quality draft assemblies of mammalian genomes from massively parallel sequence data. Proc Natl Acad Sci U S A *108*, 1513–8.

[Grabherr et al., 2011] Grabherr, M., Haas, B., Yassour, M., Levin, J., Thompson, D., Amit, I., Adiconis, X., Fan, L., Raychowdhury, R., Zeng, Q., Chen, Z., Mauceli, E., Hacohen, N., Gnirke, A., Rhind, N., di Palma, F., Birren, B., Nusbaum, C., Lindblad-Toh, K., Friedman, N. and Regev, A. (2011). Full-length transcriptome assembly from RNA-Seq data without a reference genome. Nat Biotechnol *29*, 644–52.

[Gu et al., 2008] Gu, W., Castoe, T., Hedges, D., Batzer, M. and Pollock, D. (2008). Identification of repeat structure in large genomes using repeat probability clouds. Anal Biochem *380*, 77–83.

[Henry et al., 2011] Henry, C., Overbeek, R., Xia, F., Best, A., Glass, E., Gilbert, J., Larsen, P., Edwards, R., Disz, T., Meyer, F., Vonstein, V., Dejongh, M., Bartels, D., Desai, N., D'Souza, M., Devoid, S., Keegan, K., Olson, R., Wilke, A., Wilkening, J. and Stevens, R. (2011). Connecting genotype to phenotype in the era of high-throughput sequencing. Biochim Biophys Acta .

[Hess et al., 2011] Hess, M., Sczyrba, A., Egan, R., Kim, T., Chokhawala, H., Schroth, G., Luo, S., Clark, D., Chen, F., Zhang, T., Mackie, R., Pennacchio, L., Tringe, S., Visel, A., Woyke, T., Wang, Z. and Rubin, E. (2011). Metagenomic discovery of biomass-degrading genes and genomes from cow rumen. Science *331*, 463–7.

[Jackson et al., 2009] Jackson, B., Schnable, P. and Aluru, S. (2009). Parallel short sequence assembly of transcriptomes. BMC Bioinformatics *10 Suppl 1*, S14.

[Kingsford et al., 2010] Kingsford, C., Schatz, M. and Pop, M. (2010). Assembly complexity of prokaryotic genomes using short reads. BMC Bioinformatics *11*, 21.

[Knuth, 1975] Knuth, D. (1975). The art of computer programming, vol. 1,. Addison-Wesley.

[Li et al., 2010] Li, P., Ponnala, L., Gandotra, N., Wang, L., Si, Y., Tausta, S., Kebrom, T., Provart, N., Patel, R., Myers, C., Reidel, E., Turgeon, R., Liu, P., Sun, Q., Nelson, T. and Brutnell, T. (2010). The developmental dynamics of the maize leaf transcriptome. Nat Genet *42*, 1060–7.

[Llewellyn and Eisenberg, 2008] Llewellyn, R. and Eisenberg, D. (2008). Annotating proteins with generalized functional linkages. Proc Natl Acad Sci U S A *105*, 17700–5.

[Markowitz et al., 2008] Markowitz, V., Ivanova, N., Szeto, E., Palaniappan, K., Chu, K., Dalevi, D., Chen, I., Grechkin, Y., Dubchak, I., Anderson, I., Lykidis, A., Mavromatis, K., Hugenholtz, P. and Kyrpides, N. (2008). IMG/M: a data management and analysis system for metagenomes. Nucleic Acids Res *36*, D534–8.

[Miller et al., 2011] Miller, C., Baker, B., Thomas, B., Singer, S. and Banfield, J. (2011). EMIRGE: reconstruction of full-length ribosomal genes from microbial community short read sequencing data. Genome Biol *12*, R44.

[Miller et al., 2010] Miller, J., Koren, S. and Sutton, G. (2010). Assembly algorithms for next-generation sequencing data. Genomics *95*, 315–27.

[Mortazavi et al., 2010] Mortazavi, A., Schwarz, E., Williams, B., Schaeffer, L., Antoshechkin, I., Wold, B. and Sternberg, P. (2010). Scaffolding a Caenorhabditis nematode genome with RNA-seq. Genome Res *20*, 1740–7.

[Novak et al., 2010] Novak, P., Neumann, P. and Macas, J. (2010). Graph-based clustering and characterization of repetitive sequences in next-generation sequencing data. BMC Bioinformatics *11*, 378.

[Peng et al., 2011] Peng, Y., Leung, H., Yiu, S. and Chin, F. (2011). Meta-IDBA: a de Novo assembler for metagenomic data. Bioinformatics *27*, i94–i101.

[Pennisi, 2011] Pennisi, E. (2011). Human genome 10th anniversary. Will computers crash genomics? Science *331*, 666–8.

[Pevzner et al., 2001] Pevzner, P., Tang, H. and Waterman, M. (2001). An Eulerian path approach to DNA fragment assembly. Proc Natl Acad Sci U S A *98*, 9748–53.

[Pickrell et al., 2010] Pickrell, J., Marioni, J., Pai, A., Degner, J., Engelhardt, B., Nkadori, E., Veyrieras, J., Stephens, M., Gilad, Y. and Pritchard, J. (2010). Understanding mechanisms underlying human gene expression variation with RNA sequencing. Nature *464*, 768–72.

[Pignatelli and Moya, 2011] Pignatelli, M. and Moya, A. (2011). Evaluating the fidelity of de novo short read metagenomic assembly using simulated data. PLoS One *6*, e19984.

[Qin et al., 2010] Qin, J., Li, R., Raes, J., Arumugam, M., Burgdorf, K., Manichanh, C., Nielsen, T., Pons, N., Levenez, F., Yamada, T., Mende, D., Li, J., Xu, J., Li, S., Li, D., Cao, J., Wang, B., Liang, H., Zheng, H., Xie, Y., Tap, J., Lepage, P., Bertalan, M., Batto, J., Hansen, T., Paslier, D. L., Linneberg, A., Nielsen, H., Pelletier, E., Renault, P., Sicheritz-Ponten, T., Turner, K., Zhu, H., Yu, C., Li, S., Jian, M., Zhou, Y., Li, Y., Zhang, X., Li, S., Qin, N., Yang, H., Wang, J., Brunak, S., Dore, J., Guarner, F., Kristiansen, K., Pedersen, O., Parkhill, J., Weissenbach, J., Bork, P., Ehrlich, S. and Wang, J. (2010). A human gut microbial gene catalogue established by metagenomic sequencing. Nature *464*, 59–65.

[Robertson et al., 2010] Robertson, G., Schein, J., Chiu, R., Corbett, R., Field, M., Jackman, S., Mungall, K., Lee, S., Okada, H., Qian, J., Griffith, M., Raymond, A., Thiessen, N., Cezard, T., Butterfield, Y., Newsome, R., Chan, S., She, R., Varhol, R., Kamoh, B., Prabhu, A., Tam, A., Zhao, Y., Moore, R., Hirst, M., Marra, M., Jones, S., Hoodless, P. and Birol, I. (2010). De novo assembly and analysis of RNA-seq data. Nat Methods *7*, 909–12.

[Schadt et al., 2010] Schadt, E., Turner, S. and Kasarskis, A. (2010). A window into third-generation sequencing. Hum Mol Genet *19*, R227–40.

[Schatz et al., 2010] Schatz, M., Langmead, B. and Salzberg, S. (2010). Cloud computing and the DNA data race. Nat Biotechnol *28*, 691–3.

[Simpson and Durbin, 2010] Simpson, J. and Durbin, R. (2010). Efficient construction of an assembly string graph using the FM-index. Bioinformatics *26*, i367–73.

[Simpson et al., 2009] Simpson, J., Wong, K., Jackman, S., Schein, J., Jones, S. and Birol, I. (2009). ABySS: a parallel assembler for short read sequence data. Genome Res *19*, 1117–23.

[Sogin et al., 2006] Sogin, M., Morrison, H., Huber, J., Welch, D. M., Huse, S., Neal, P., Arrieta, J. and Herndl, G. (2006). Microbial diversity in the deep sea and the underexplored 'rare biosphere'. Proc Natl Acad Sci U S A *103*, 12115–20.

[Sun et al., 2011] Sun, S., Chen, J., Li, W., Altintas, I., Lin, A., Peltier, S., Stocks, K., Allen, E., Ellisman, M., Grethe, J. and Wooley, J. (2011). Community cyberinfrastructure for Advanced Microbial Ecology Research and Analysis: the CAMERA resource. Nucleic Acids Res *39*, D546–51.

[Tang et al., 2009] Tang, F., Barbacioru, C., Wang, Y., Nordman, E., Lee, C., Xu, N., Wang, X., Bodeau, J., Tuch, B., Siddiqui, A., Lao, K. and Surani, M. (2009). mRNA-Seq whole-transcriptome analysis of a single cell. Nat Methods *6*, 377–82.

[Treangen et al., 2011] Treangen, T., Sommer, D., Angly, F., Koren, S. and Pop, M. (2011). Next generation sequence assembly with AMOS. Curr Protoc Bioinformatics *Chapter 11*, Unit 11.8.

[Tringe and Rubin, 2005] Tringe, S. and Rubin, E. (2005). Metagenomics: DNA sequencing of environmental samples. Nat Rev Genet *6*, 805–14.

[Tyson et al., 2004] Tyson, G., Chapman, J., Hugenholtz, P., Allen, E., Ram, R., Richardson, P., Solovyev, V., Rubin, E., Rokhsar, D. and Banfield, J. (2004). Community structure and metabolism through reconstruction of microbial genomes from the environment. Nature *428*, 37–43.

[Venter et al., 2004] Venter, J., Remington, K., Heidelberg, J., Halpern, A., Rusch, D., Eisen, J., Wu, D., Paulsen, I., Nelson, K., Nelson, W., Fouts, D., Levy, S., Knap, A., Lomas, M., Nealson, K., White, O., Peterson, J., Hoffman, J., Parsons, R., Baden-Tillson, H., Pfannkoch, C., Rogers, Y. and Smith, H. (2004). Environmental genome shotgun sequencing of the Sargasso Sea. Science *304*, 66–74.

[von Mering et al., 2007] von Mering, C., Hugenholtz, P., Raes, J., Tringe, S., Doerks, T., Jensen, L., Ward, N. and Bork, P. (2007). Quantitative phylogenetic assessment of microbial communities in diverse environments. Science *315*, 1126–30.

[Woyke et al., 2010] Woyke, T., Tighe, D., Mavromatis, K., Clum, A., Copeland, A., Schackwitz, W., Lapidus, A., Wu, D., McCutcheon, J., McDonald, B., Moran, N., Bristow, J. and Cheng, J. (2010). One bacterial cell, one complete genome. PLoS One *5*, e10314.

[Zerbino and Birney, 2008] Zerbino, D. and Birney, E. (2008). Velvet: algorithms for de novo short read assembly using de Bruijn graphs. Genome Res *18*, 821–9.

[Zerbino et al., 2009] Zerbino, D., McEwen, G., Margulies, E. and Birney, E. (2009). Pebble and rock band: heuristic resolution of repeats and scaffolding in the velvet short-read de novo assembler. PLoS One *4*, e8407.