TeraGrid for Genome

Assembly, Annotation and Analysis

Don Gilbert, Indiana University

ABSTRACT

Motivation: A general problem in bioinformatics is enabling use of large biology data sets on shared cyberinfrastructure. Parallelizing data access rather than applications has potential for effective Grid use of existing and new biology analyses.

Results: New insect and crustacean genomes have been analyzed on TeraGrid to assess data grid methods in genome informatics. Rapid Grid analyses have facilitated rapid biology discoveries in these genomes. Data grid methods can automate genome partitioning and parallel processing with many genome tools.

Availability: http://insects.eugenes.org/DroSpeGe/

Contact: Don Gilbert, gilbertd@indiana.edu

INTRODUCTION

Most bioinformatics applications are "embarassingly parallel" --- that is they consist of doing thousands of independent computations ... The real problems in computer architecture for bioinformatics have to do with handling the data, not the computation. We need high-performance data architectures... (Kevin Karplus, www.bio.net/mm/comp-bio/2006-April/)

A common, ongoing task for research with genome databases is to compare an organism's genome and proteome with related species, and other sequence data (ESTs, SNPs, transposable elements). This requires significant computational infrastracture, where reusable tools, protocols and resources can be valuable. Public genome archives exceed one billion sequence traces from over 1,000 organisms (NCBI, 2006). This number will increase rapidly as costs decline and science uses for genomes increase (e.g. Siepel *et al*. 2005). Good software to fully assembly, analyze and compare genomes are available now, but the ability to employ these tools is limited to those with extensive computational resources.

Genome software including **gene finders**, **homology comparison**, **multiple alignment** tools, and **phylogenetic comparison** often work on subsets of genome sequence, and collate results. Versions of BLAST have been parallelized for Grid and cluster computing, but effort to parallelize others trail behind the development of new tools. Promising newer genome tools draw data from several sources: cross-species homologies, large scale functional and interaction data and primary genome sequences. A practice common in genome analyses is ad hoc development of data processing scripts to split and collate genome data and results. This can be automated for Grid computing. Depending on the analysis, splitting and collation operations can be handled in a generic manner.

Analyses of invertebrate genomes

Assessment of TeraGrid to analyze new invertebrate genomes has been performed in the context of uses for the genome database community. This assessment includes newly sequenced genomes for *Daphnia pulex* and twelve *Drosophila* genomes. Genome database tools from the Generic Model Organism Database (GMOD, Stein et al., 2002) project are used to organize TeraGrid results for public access.

For each of *Daphnia* and twelve *Drosophila* genomes, a comparison is made to nine proteomes, with 217,000 proteins, drawn from source genome databases, Ensembl and NCBI. These reference proteomes are human, mouse, zebrafish, fruitfly, mosquito, bee, worm, mustard weed, and yeast. Sizes of the new genomes are in the 150 Mb to 250 Megabase range. Protein-genome DNA alignment is performed with tBLASTn, using a Grid (MPI parallel) version developed at Indiana University Technology Services. A TeraGrid run for each genome takes 18 hours using 64 processors. Whole genome DNA-DNA genome alignments are performed also. Gene predictions with SNAP (Korf, 2004) are generated. Over the course of 6 months, with 2 to 3 genome assembly updates per species, and error corrections, the total TeraGrid 64-cpu usage per genome has been approximately 4 days.

Public access to this research, in the form of genome maps (GBrowse), similarity searches (web BLAST), data mining (BioMart), along with genome summaries, are provided at web databases **wfleabase.org** (Daphnia) and **insects.euGenes.org** (Drosophila). This work has enabled many bioscientists to have rapid, usable access to the new genomes, facilitating new discoveries and understanding of the evolution, comparative biology, and genomics of these model organisms.

Genomes Compared

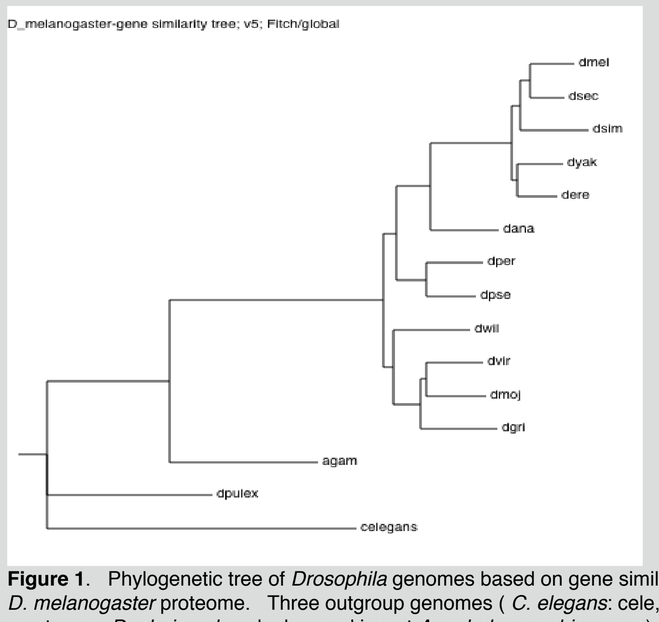

Figure 1. Phylogenetic tree of *Drosophila* genomes based on gene similarity to *D. melanogaster* proteome. Three outgroup genomes (*C. elegans*: cele, crustacean *Daphnia pulex*: dpulex, and insect *Anopholes gambia*: agam) are included with *Drosophila* (Dmel to Dgri). Distance matrices were computed from BLAST scores (gene similarity) and adjacent gene pairs (gene order) using R, phylogenies were computed with PHYLIP:Fitch from distances, and drawn with Phylodendron. These trees for the most part match the accepted phylogenies. *D. simulans* is an outlier, its assembly is a mosaic from several populations, which may be affecting its placement. The *D. willistoni* placement differs somewhat from accepted phylogeny, and from the gene order placement.

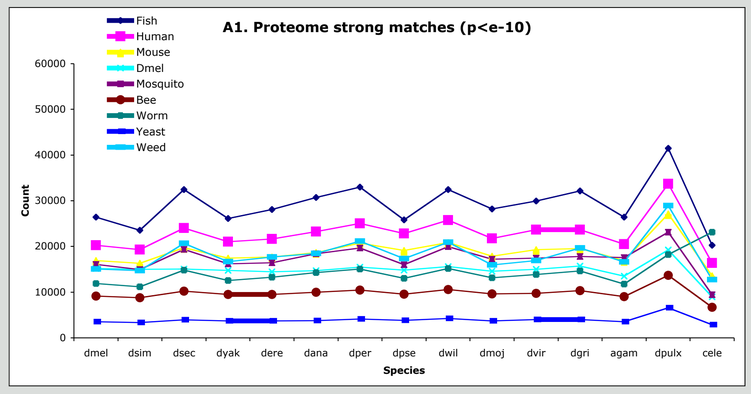

Figure 3. Gene matches in new invertebrate genomes. tBLASTn is used to match query proteomes to target genomes. Target genomes are twelve *Drosophila* species (Dmel .. Dgri), the mosquito *Anopholes gambia* (Agam), the crustatean *Daphnia pulex* (Dpulx), and worm *C. elegans* (Cele). These counts include many duplicate matches, to different as well as same genome locations.

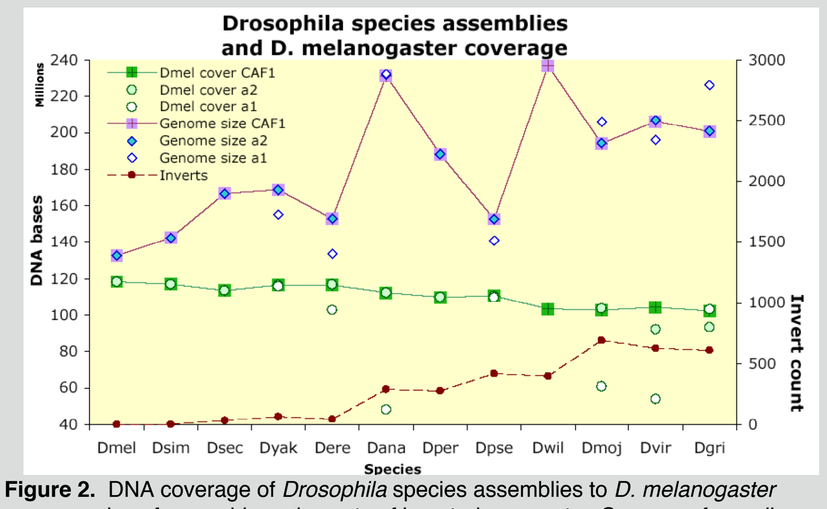

Figure 2. DNA coverage of *Drosophila* species assemblies to *D. melanogaster* genome, size of assembly and counts of inverted segments. Coverage for earlier assemblies along with latest assemblies (CAF1) are shown.

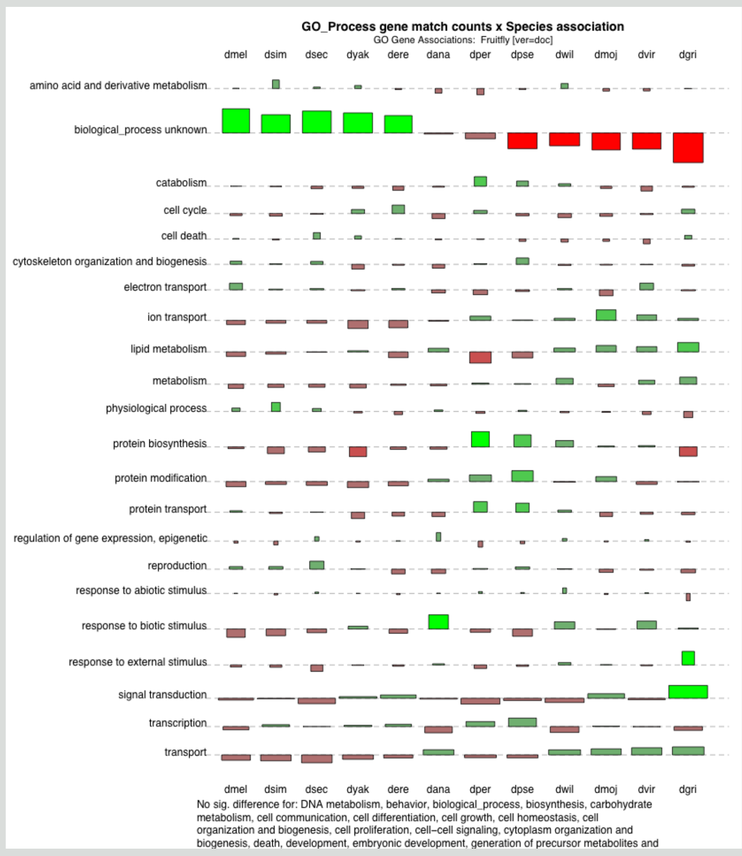

Figure 5. Putative gene gain and loss among *Drosophila* species in Gene Ontology biological process groupings. These may indicate where species genes differ in functional categories. Brighter colors indicate greater statistical significance. Low counts may be due to divergence rather than lack. High-scoring Segment Pair (HSP) groupings are scored, and include various events: gene duplications, alternate splice exons within genes, new genes that appear composed of exons from other genes, as well as computational artifacts.

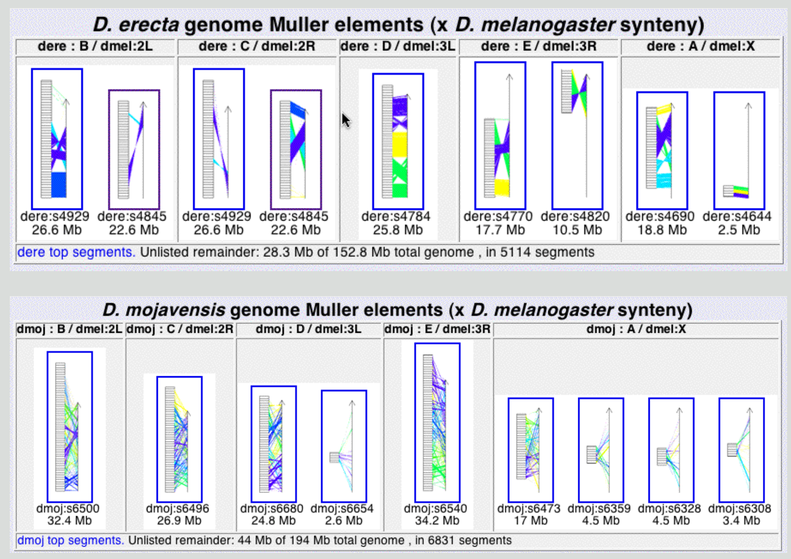

Figure 4 (A,B). Large scale synteny of two Drosophila genomes to *D. melanogaster* genome, showing conserved chromosome arms (Muller elements). Figure A is *D. erecta*, a near relative to Dmel, and Figure B is *D. mojavensis*, distant from Dmel. Each chromosome picture has the query genome on the left as a "ladder", and the synteny source on the right as an arrow. Between are colored lines of high matching regions (colors distinguish clusterings).

TeraGrid Basics

The assessment shows that use of TeraGrid for shared genome computations is feasible. This use of TeraGrid for genome-sized computations has aided the NIH-sponsored *Drosophila* species sequencing with a quick assessment of assembly qualities for gene annotation. Improved assemblies have more gene matches, and fewer duplicate matches. Hurdles to wider use of TeraGrid by genome informaticians include a large learning and setup cost in time and effort. Failed runs were a significant portion (TeraGrid outages, software and data errors), and required personal attention to correct. The major part of the human effort involves preparing data, distributing to grid nodes, retrieving volumes of results, and combining and summarizing those. It is this aspect where data grid tools can facilitate uses of TeraGrid and Grid resources for genome informatics.

Table 1. TeraGrid usage steps.

Step	*Notes*
Preparation	*One time*
1. Obtain TeraGrid account	Via web http://www.teragrid.org/userinfo/
2. Establish certificates	Grid-security entries; test proxy; local workstation certificate
3. Locate biology software	Find and compile parallel applications
Processing	*Per analysis*
4. Locate and prepare data	partition, shred & randomize
5. Transfer data to TeraGrid	FTP, secure-shell, other
6. Configure and run analysis	Globus run scripts, attention to errors, queuing
7. Return and collate results	Post-process to combine results from nodes; e.g. to-GFF for map view of genome blast.

Basic steps as shown in Table 1 for using TeraGrid for genome data are not complicated, but require learning and trial and error for the new user. Difficulties in these steps are being addressed by TeraGrid developers. Some of these can be streamlined for specific needs of genome informatics. Of these, steps 4 to 7 represent bioinformatics needs. Data selection, preparation, transport to TeraGrid, and return of results, in collated form, to the scientist are the special needs. Methods for step 6 are in the realm of workflow tools developed elsewhere and applied in this project.

Genome Data Grid methods

Many genome computations work iteratively over the genome base "string", where genome substrings or contigs can be effectively analyzed independently then results collated. One major exception to this is genome assembly, which requires analyses of all genome fragments at once. Data grid methods to partition and analyze in parallel a genome can be implemented with a few steps to locate data, copy to grid, and return results:

1. @virtualdata= biodirectory("find protein coding sequences for *Drosophila* species"), as an example from a wide range of queries.
2. @realdata= biodirectory("get locators for @virtualdata split *n* ways"), for *n* compute nodes
3. for i (1.. *n*) { copy(realdata[i],gridcpu[i]); results[i]=runapp(gridcpu[i]) }
4. result_table = collate(@results);

These steps summarize actions to find/query data directories, copy subsets to distributed computing nodes, and return results from the analyses, collated from the compute nodes. Steps 2, 3 are the core of a data-grid system. Step 3 means that analysis applications need not have any special data access methods. Data grid tools can transport appropriate data parts to each compute node. Steps 1,4 may be separate systems. E.g. any database query system could work for step 1 as long as it returned IDs usable for selecting subsets. Step 4 would include many tools that assemble and summarize raw results, such as those from BioPerl. These steps should be overseen by a workflow system capable at data and compute tasks. Genome partitioning has been tested during preliminary annotation of genomes, and is equally effective to MPI-parallelized BLAST. Data partitioning also permits parallel analyses with non-parallelized applications, such as gene finding, multiple alignment and orthology analysis.

References

Colbourne, J.K., Singan, V.R., Gilbert, D.G. 2005. wFleaBase: the Daphnia genome database, **BMC Bioinformatics**, 6:45 doi:10.1186/1471-2105-6-45 URL: wfleabase.org

Korf, Ian 2004. Gene finding in novel genomes. **BMC Bioinformatics** 2004, 5:59 URL: http://www.biomedcentral.com/1471-2105/5/59

Stein L.D., Mungall, C., Shu, S., Caudy, M., Mangone, M., Day, A., Nickerson, E., Stajich, J.E., Harris, T.W., Arva, A., and Lewis, S. 2002. The generic genome browser: a building block for a model organism system database. **Genome Res.** 12: 1599-610. URL: www.gmod.org