


EMBnet.journal

Volume 17 Nr. 2

December 2011

- 
- A photograph of a zebra and its foal in a savanna landscape. The adult zebra is in the foreground, facing the camera, with its distinctive black and white stripes. The foal is behind it, also facing the camera. The background is a vast, open plain with dry grass and a few distant trees under a clear sky.
- **BioVel: Biodiversity Virtual e-Laboratory**
 - **Taxonomic Assignment in Metagenomics with TANGO**
 - **Superclusteroid: a Web tool dedicated to data processing of protein-protein interaction networks and more...**

Editorial

During the last twenty years, EMBnet has evolved from a collaborative European network into a global organisation, with representatives also from Asia, Africa, Australia and the Americas. Embracing its growing global dimension, EMBnet is now committed to fostering bioinformatics training and to disseminating bioinformatics skills throughout its member countries.

This global activity is reflected in the article describing the "ISCB Africa ASBCB Conference on Bioinformatics," where EMBnet participated actively. The meeting, held in South Africa, was the first time that EMBnet officially sponsored an event of this kind. This conference was an important opportunity to synergise with both the African Society for Bioinformatics and Computational Biology (ASBCB) and the International Society for Computational Biology (ISCB).

Next year, EMBnet will be present at events taking place on several different continents: among others, EMBnet will be present at, and will sponsor, the ISCB Latin America 2012 "Conference on Bioinformatics" in Santiago de Chile (17-21 March, 2012), and the "International Workshops on Bioinformatics – 2012", to be held 16-27 January, 2012 in the Center for Genomic Sciences facilities in Cuernavaca Morelos, México. These events will be fully covered in reports in forthcoming issues of EMBnet.journal.



Protein Spotlight (ISSN 1424-4721) is a periodical electronic review from the SWISS-PROT group of the Swiss Institute of Bioinformatics (SIB). It is published on a monthly basis and consists of articles focused on particular proteins of interest. Each issue is available, free of charge, in HTML or PDF format at <http://www.expasy.org/spotlight>.

We provide the EMBnet community with a printed version of issue 132. Please let us know if you like this inclusion.

We strongly encourage you, and all our readers, to write to us and propose other meetings or events that might also benefit from being covered, and to use our Open Journal system to [submit](#)¹ your own reports.

EMBnet.journal Editorial Board

Contents

Editorial	2
News	
Announcement from the EMBnet Associate Node in South Africa: UCT and CPGR join forces with international Pharmacogenomics Initiative focusing on African diseases	3
BioVel: Biodiversity Virtual e-Laboratory	5
Reports	
ISCB Africa ASBCB Conference on Bioinformatics and eBioKit Workshop	7
Research Papers	
Superclusteroid: a Web tool dedicated to data processing of protein-protein interaction networks ...	10
Taxonomic Assignment in Metagenomics with TANGO	16
Using the Grid to run population dynamics simulations	21
Protein Spotlight	29
Node information	31

Editorial Board:

Erik Bongcam-Rudloff, Department of Animal Breeding and Genetics, SLU, SE, erik.bongcam@slu.se

Teresa K. Attwood, Faculty of Life Sciences and School of Computer Sciences, University of Manchester, UK, teresa.k.attwood@manchester.ac.uk

Domenica D'Elia, Institute for Biomedical Technologies, CNR, Bari, IT, domenica.delia@ba.itb.cnr.it

Andreas Gisel, Institute for Biomedical Technologies, CNR, Bari, IT, andreas.gisel@ba.itb.cnr.it

Laurent Falquet, Swiss Institute of Bioinformatics, Génopode, Lausanne, CH, Laurent.Falquet@isb-sib.ch

Pedro Fernandes, Instituto Gulbenkian. PT, pfern@igc.gulbenkian.pt

Lubos Klucar, Institute of Molecular Biology, SAS Bratislava, SK, klucar@EMBnet.sk

Martin Norling, Swedish University of Agriculture, SLU, Uppsala, SE, martin.norling@slu.se

Announcement from the EMBnet Associate Node in South Africa

UCT and CPGR join forces with international Pharmacogenomics Initiative focusing on African diseases



Reinhard Hiller¹, Raj Ramesar²

¹Centre for Proteomic and Genomic Research (CPGR), South Africa

²University of Cape Town, South Africa

Cape Town, South Africa, 17 June 2011

The Division of Human Genetics at the University of Cape Town (UCT) and the Centre for Proteomic and Genomic Research (CPGR), Cape Town, South Africa, proudly announce that they will be joining the 'Pharmacogenomics for Every Nation Initiative' (PGENI). Jointly, the two parties will form a South African PGENI Centre of Competence for conducting translational research relevant to the local burden of disease and to the most appropriate drugs for treating diseases in African populations.

The aim of the PGENI Centre will be to conduct large-scale studies investigating the prevalence of specific genetic traits (single-nucleotide polymorphisms, or SNPs) in South African populations, and the relationship of such traits with drug efficacy. Side-effects in drug treatments are a major concern for health-care providers worldwide. However, they present a particular problem in developing nations for two reasons: (i) most drugs available today have been developed for use in Caucasian populations, and have not been tailored to the genetic make-up of other population groups; (ii) drug side-effects create a significant financial burden for health-care systems in developing nations, where provision of effective treatments is critical for tackling the burden of disease.

The South African PGENI Centre will initially concentrate on investigating the prevalence of SNPs with known implications in drug efficacy. In order to do this, the Centre will use the Affymetrix DMET™ Plus application in cross-sectional pharmacogenomic studies. Following an initial pilot study, where a few hundred samples will be analysed, the Centre's aim is eventually to generate data-sets from thousands of individuals. These, in conjunction with bio-computational data-mining, will be used to determine drug-specific SNP profiles, and to develop recommendations for policy makers and health-care providers to improve the efficacy of drug treatments in South Africa.

According to Raj Ramesar, Professor of Human Genetics at UCT, and Scientific Director of the PGENI Centre in South Africa: "Our focus is on using powerful genomic tools to understand the exact mechanistic processes that lead to disease. This approach then leads one to devise new generations of drugs and therapeutics, which are better targeted to relevant points of biological interest in the disease process. However, different individuals process drugs at different rates, as much as they process foods and nutrients at different rates; and these processes and rates are genetically determined. We generally import drugs from international vendors, and use them to treat symptoms or diseases for which we presume biological processes are the same between our populations and where the drugs were originally manufactured and trialled. The work we plan to undertake in large numbers of African populations aims to optimise drug use for specific diseases, according to an individual's ability to process such drugs optimally."

"We are pleased to use our expertise in conducting large-scale genomic studies in a joint effort with UCT and PGENI, aimed at improving the efficacy of drug treatments in South Africa", said Reinhard Hiller, Managing Director of the CPGR. "The DMET™ Plus is an application very well suited to generating high-quality pharmacogenetic data-sets. Being able to use this tool to unravel genetic information that can be used to improve the quality of health-care in local populations will ensure that the project's scientific objectives will be met. What's more, we will be able to translate findings into practical applications with a tangible benefit for the community. South Africa is seeking to strengthen its capabilities in genetics

and genomics related to health, and this program will form a significant step in this direction."

Dr. Howard McLeod, Director of PGENI, based at the University of North Carolina in the USA, commented: "We are pleased that UCT and CPGR are bringing their extensive expertise to PGENI. We need partners that have the rare ability to perform high-quality science and guide policy development, and have found those skills in UCT/CPGR. As long-standing leaders in Africa, the Cape Town team have had immediate impact on shaping high-impact PGENI strategies for improving the selection of medications for African countries and beyond."

About the Division of Human Genetics at UCT

The Division of Human Genetics at UCT concentrates on clinical service delivery, through medical genetics clinics at affiliated hospitals. Medical Genetics services are supported by molecular and cytogenetic diagnostic laboratories. The Division has a greater reach within the clinical environment through its MRC Research Unit for Human Genetics, which focuses on the genetic basis of a wide range of the common non-communicable diseases. These contribute a significant burden of disease in South Africa and continentally. The division's more recent attention to genomic variation in indigenous African populations has been important in relating such variations to disease predisposition and variations in response to therapeutics. More information on research in the Division of Human Genetics is available at: www.uct.ac.za/depts/genetics. For more information about the Human Genetics Division, please contact: Professor Raj Ramesar at Raj.Ramesar@uct.ac.za, or:



Patricia Lucas
Tel: (021) 650 5428
Cell: 076 292 8047
E-mail: pat.lucas@uct.ac.za
University of Cape Town
Website: www.uct.ac.za

About the CPGR

The CPGR is a specialist not-for-profit contract research organisation established in South Africa to provide support and services to the life science and biotech communities, based on an initiative by the Department of Science and Technology (DST) to boost the development of a bio-econ-



omy in South Africa. The organisation, based in Cape Town, combines state-of-the-art information-rich genomic and proteomic ('omics') technologies with bio-computational pipelines, and biological models, to create unique solutions in the human health and agri-biotech sectors. The CPGR is funded by the Technology Innovation Agency (TIA) in South Africa. Please visit www.cpgr.org.za for more information or contact Dr. Reinhard Hiller (reinhard.hiller@cpgr.org.za) with specific requests.

About PGENI

The Pharmacogenomics for Every Nation Initiative (PGENI) is an enterprise of the Institute of Pharmacogenomics and Individualised Therapy (IPIT) at the University of North Carolina (UNC). PGENI works to integrate genetic-risk data for an individual country and World Health Organisation essential-medicine recommendations into public-health decision-making without placing an extra burden on health-care funding and technology infrastructure. PGENI has regional centres in Brazil, Jordan, South Africa, India, China, Mexico and Ghana, and is active in more than 100 countries. For more information, please visit <http://pgeni.unc.edu/>.

IPIT is an initiative of the UNC Eshelman School of Pharmacy, in collaboration with the UNC School of Medicine, UNC Gillings School of Global Public Health, and the School of Nursing, with substantial support from the Lineberger Comprehensive Cancer Centre and the Carolina Centre for Genome Sciences. The mission of IPIT is to employ an interdisciplinary approach to tailor therapies and enable the delivery of individualised medical practice. IPIT also offers the services of facilities in molecular genomics, cellular phenotyping and pharmacoinformatics to add to the excellent core facilities already existing at UNC. For more information, please visit: <http://ipit.unc.edu/>.

BioVeL: Biodiversity Virtual e-Laboratory



Saverio Vicario¹, Alex Hardisty², Niobe Haitas³

¹CNR - Institute for Biomedical Technologies, Italy

²Cardiff University, United Kingdom

³HealthGrid, France

E-solutions for the management of biodiversity in the 21st century

Scientists are being pressured to provide convincing evidence of changes to contemporary biodiversity, to identify factors causing decline in biodiversity and to predict the impact of, and

to suggest ways of combating, biodiversity loss. Altered species distributions, the changing nature of ecosystems and increased risks of extinction all have impacts in important areas of societal concern. Biologists and environmental scientists are asked to provide decision support for managing the biodiversity component of our environment at multiple scales (genomic, organism, habitat, ecosystem, landscape) to prevent and mitigate such losses. Generating the evidence and providing decision support relies, increasingly, on large collections of data held in digital formats, and the application of substantial computational capability and capacity to analyse and model such data, and to run simulations.

The BioVeL approach

BioVeL, a 3-year FP7 project, aims to catalyse the energy and knowledge present in the research community, helping to address the challenge of understanding and managing biodiversity. More precisely, the goal of the BioVeL project is to provide a seamlessly connected informatics

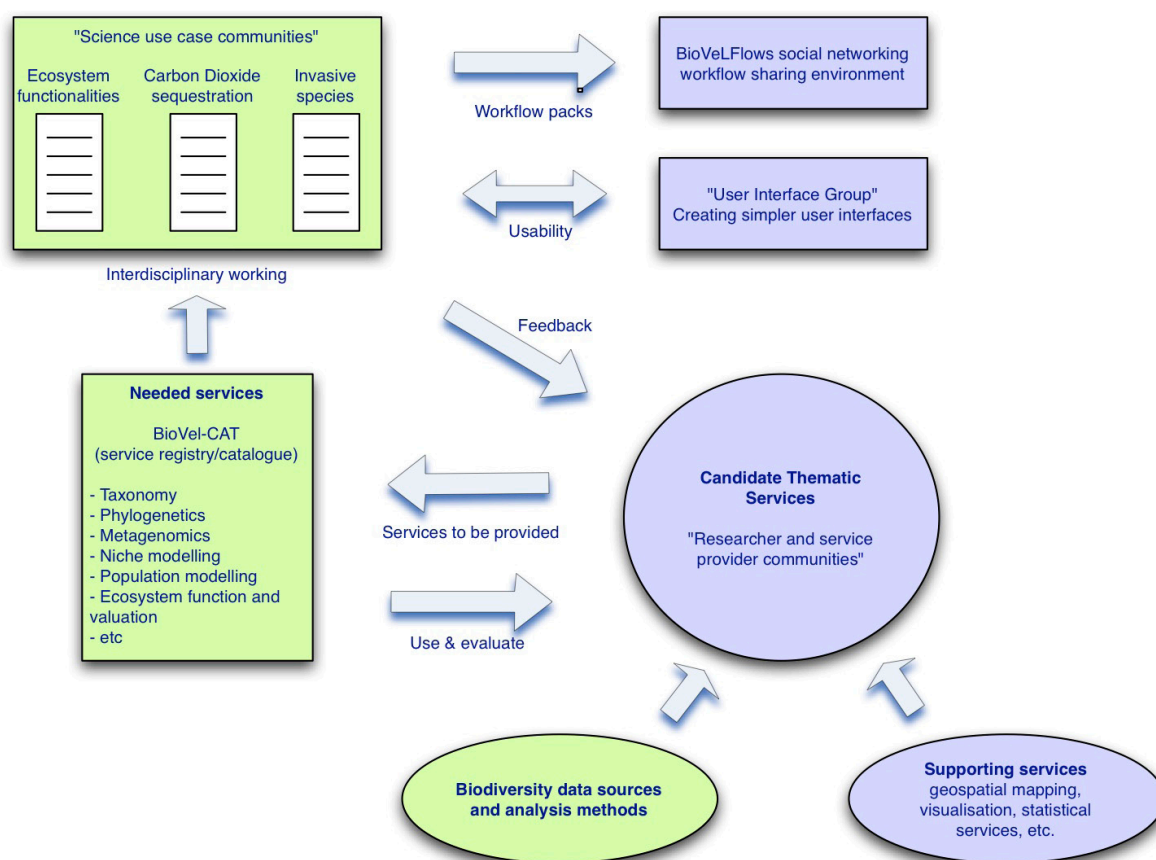


Figure 1. Algorithm selection module of the Superclusteroid tool.

environment that makes it easier for biodiversity scientists both to carry out *in silico* analysis of relevant biodiversity data, and to pursue *in silico* experiments based on the composition and execution of complex digital data-manipulation and -modelling tasks. In September, the Biodiversity Virtual e-Laboratory (BioVeL) project had its kick-off. In BioVeL scientists and technologists will work together to meet the needs and demands for 'e-Science', and to create a production-quality informatics infrastructure to enable pipelining of data and analysis into efficient, integrated workflows. Workflows represent a way of speeding up scientific advance when that advance is based on the manipulation of digital data (Gil *et al.*, 2007).

BioVeL will not produce new applications or software, but will help developers to expose useful software programs as Web services. It will allow users to access and compose them in workflows, and to share and comment the composed workflows, using an adaptation of the [myGrid](http://www.mygrid.org.uk/)¹ suite. Furthermore, a large section of the project is dedicated to engaging the community along three lines of action: 1) following the myGrid paradigm, all services and workflows will be inserted within a social network framework that will ensure feedback and quality control of best practice; 2) the project will designate "pals", who are persons knowledgeable in a specific scientific field, who will make the connection between the ICT part of the project and the user community; 3) a series of workshops will ensure more formal occasions to engage with, and collect feedback from, the community of users and developers. To allow a fast reaction cycle between input from the community and ICT developers, the project will unfold using an agile-process paradigm. To seed the infrastructure with the first workflows, the project will focus on three science use cases from the biodiversity community (ecosystems services, CO₂ sequestration, invasive species management), and the workflows will be built taking applications from the following areas of knowledge: taxonomy, phylogenetics, metagenomics, ecological niche modelling, ecological population modelling, ecosystem functioning and valuation. For each of these areas, the project will nominate a "pal". The "pal" will help to compose the first workflows, together with the experts of the three focal questions, and ensure its initial correct use.

On the top of these six areas of knowledge there will be a geospatial set, to allow integration of biological applications in a spatial context (*i.e.*, biogeography or phylogeography). Furthermore, to handle all the problems linked to format and congruity of newly-formed workflows, a shim service set taken from [Taverna](http://www.taverna.org.uk/)² and the [EDAM ontology](http://edamontology.org/)³ will be used. Figure 1 shows a conceptual scheme of BioVeL.

We invite all interested developers to become friends of BioVeL, to participate in the workshops and to follow the progress of the project from now onwards. We think that a large social component of the project will both facilitate interaction and feedback among developers and user scientists, and increase the impact of research on society.

References

1. Gil Y, Deelman E, Ellisman M, Fahringer T, Fox G, Gannon D, Goble C, Livny M, Moreau L, Myers J (2007) Examining the Challenges of Scientific Workflows, *Computer* **40**, 24-32.

¹ <http://www.mygrid.org.uk/>

² <http://www.taverna.org.uk/>

³ <http://edamontology.sourceforge.net/>

ISCB Africa ASBCB Conference on Bioinformatics and eBioKit Workshop



**Etienne de Villiers¹, Judit Kumuthini²,
Erik Bongcam-Rudloff³**

¹ILRI Bioinformatics group, Kenya

²Centre for Proteomic and Genomic Research (CPGR), South Africa

³Swedish University of Agricultural Sciences, Uppsala, Sweden

The International Society for Computational Biology (ISCB¹) and the African Society for Bioinformatics and Computational Biology (ASBCB²) held the ISCB Africa ASBCB Conference on Bioinformatics in Cape Town, South Africa, in March 2011. The meeting constituted the second joint meeting of ISCB and ASBCB, and the third conference of the ASBCB on the Bioinformatics of African Pathogens, Hosts and Vectors. The conference was preceded by a two-day workshop at the [University of the Western Cape](http://www.uwc.ac.za)³. ASBCB is a society dedicated to the advancement of bioinformatics and computational biology in Africa.

The society works with the [ISCB African Regional Student Groups](http://www.iscbsc.org/content/regional-student-groups)⁴ to provide training courses and a mentorship programme, to help train the current and next generation of African bioinformatics students.

EMBnet was very well represented at the conference, both as one of the official sponsors, and in providing two trainers for the preceding workshop. EMBnet had an exhibition stand with promotional material, including copies of EMBnet journal and its new promotional pamphlet, organised by Judit Kumuthini from the Centre for Proteomic and Genomic Research (CPGR⁵), and the EMBnet node in South Africa. Displayed for the first time was a promotional poster that is now being presented at all major meetings, courtesy of EMBnet's Publicity & Public Relations Project Committee.

Many visitors to the EMBnet stand were interested in signing up to the mailing list, indicating the continuing interest of scientists in Africa in EMBnet's activities. Several also expressed interest in establishing a dedicated node for their country or Institute.

At the stand, there was a working copy of the [eBioKit](http://collab.hgen.slu.se/software/ebiokit)⁶, developed by the research team of Erik Bongcam-Rudloff, which was very well received by attendees. eBioKit is a novel system for teaching bioinformatics in places where there is limited Internet access, and hence limited on-line access to bioinformatics software and databases. Many of these resources are installed



Participants at the eBioKit tutorials.



EMBnet stand.

1 www.iscb.org
2 www.asbcb.org
3 www.uwc.ac.za

4 www.iscbsc.org/content/regional-student-groups
5 www.cpgr.org.za
6 <http://collab.hgen.slu.se/software/ebiokit>



Cape Town.

on the eBioKit by default, including [Ensembl](#)⁷. One of the co-developers of [Jalview2](#)⁸, Dr. David Martin from [Dundee University](#)⁹, Scotland, kindly offered to install Jalview, which is therefore now also available as part of the eBioKit package.

The two-day workshop was held at SANBI ([South African Bioinformatics Institute](#)¹⁰), University of Western Cape ([UWC](#)¹¹) in Cape Town, South Africa, on the 7th and 8th of March, with three parallel sessions: the first showcased the online tools of the European Bioinformatics Institute ([EBI](#)¹²); the second introduced EMBnet's eBioKit; and the third concerned Genome Wide Association Studies (GWAS) and population genetics.

The EBI Roadshow workshop included sessions that introduced participants to databases and tools hosted at EBI, including those for sequence searching and alignment, gene expression data analysis, and interaction and pathway analysis. These popular sessions attracted 30 or more participants.

Erik Bongcam-Rudloff from the Swedish EMBnet node, Etienne de Villiers from the Kenyan



Judith Kumuthini supervising students.

[EMBnet node](#)¹³, and Judit Kumuthini and Dane Kennedy from CPGR, taught the eBioKit workshop. This attracted around 20 participants from a variety of Institutions and Universities of several different African countries, including biologists, computational biologists, geneticists and bioinformaticians. The main objectives were to: introduce the eBioKit, with its large set of commonly used bioinformatics tools and databases, and promote its use as an advanced training tool; promote the participation and training of new in-

7 www.ensembl.org

8 www.jalview.org

9 www.dundee.ac.uk

10 www.sanbi.ac.za

11 www.uwc.ac.za

12 www.ebi.ac.uk

13 <http://hpc.ilri.cgiar.org>



Workshop participants.

investigators in the field of bioinformatics; promote communication between these scientists and locally relevant bioinformatics efforts through EMBnet's activities; and strengthen the bioinformatics network in Africa via EMBnet.

Following the introduction to the eBioKit were sessions demonstrating both how to access its installed databases using [MRS](http://mrs.cmbi.ru.nl/mrs-5)¹⁴, and how to use EMBOSS/wEMBOSS (Sarachu and Colet, 2004). Judit Kumuthini and Dane Kennedy were on hand to give help during the practical demonstrations. On the second day, participants were introduced to two more resources found in the eBioKit, namely Ensembl and [Galaxy](http://main.g2.bx.psu.edu)¹⁵, a Web-based platform for data-intensive biomedical research.

Now a global bioinformatics network, part of EMBnet's mission is to foster bioinformatics training and to disseminate bioinformatics skills throughout its member countries. This was the first time that EMBnet officially sponsored an event not directly related to its member activities, but was an important opportunity to synergise with the work of the ASBCB, one that we hope to embrace again in future. In his closing speech,

Daniel Massiga, Chair of the ASBCB, warmly recognised EMBnet's role in this collaborative approach to developing education and skills in the African continent.

References

Sarachu M, Colet M (2004) wEMBOSS: a web interface for EMBOSS. *Bioinformatics* **21**, 540–541.

¹⁴ <http://mrs.cmbi.ru.nl/mrs-5>

¹⁵ <http://main.g2.bx.psu.edu>

Superclusteroid: a Web tool dedicated to data processing of protein-protein interaction networks



**Athina Ropodi^{1,2,#}, Nikolaos Sakkos^{2,#},
Charalampos Moschopoulos^{2,3}, George Magklaras⁴, Sophia Kossida^{2,*}**

¹Department of Informatics, University of Athens, GR-15784, Athens, Greece.

²Bioinformatics & Medical Informatics Team, Biomedical Research Foundation of the Academy of Athens, Soranou Efessiou 4, GR-11527, Athens, Greece.

³Department of Computer Engineering & Informatics, University of Patras, GR-26500, Rio, Greece.

⁴The Biotechnology Centre of Oslo, University of Oslo, P.O. Box 1125 Blindern, 0317 Oslo, Norway

[#]To whom correspondence should be addressed.

^{}Equal contribution to this work.*

Abstract

The study of proteins and the interactions between them, known as Protein-Protein Interactions (PPI), is extremely important in interpreting all biological cellular functions. In this article, a new web tool called Superclusteroid is presented which can analyse PPI data, in order to detect protein complexes or characterise the functionality of unknown proteins. The tool is essentially an intuitive PPI data processing pipeline. It supports various input file formats and provides services such as clustering, PPI network visualisation and protein cluster function prediction. Each Superclusteroid service can be used in a sequential manner or on an individual basis. In order to assess the reliability of our tool to infer PPIs, the results of the tool were compared to already known MIPS database complexes and a case scenario is presented where a known protein complex is predicted and the functionality of some of its proteins is revealed.

Availability: Superclusteroid is freely available online at <http://superclusteroid.uio.no/>.

Background

In the recent era, high-throughput detection methods (Ito *et al.*, 2001; Gavin *et al.*, 2002; Stoll *et al.*, 2005; Willats, 2002) have produced a vast amount of biological data to be analysed using computational methods. Proteomics is the discipline with the objective to analyse and understand all data concerning proteins. A proteome-wide approach of understanding protein function is very important, as it is widely known that proteins rarely act alone at a biochemical level and they interact with other proteins (Bu *et al.*, 2003). This type of protein-protein interactions can easily be described as a protein-protein interaction network (PPI network), where the nodes represent proteins and the edges the interactions among them.

As protein-protein interactions are a crucial part of cellular processes, it is understandable that the processing of large-scale experiment data is extremely useful. In fact, the identification of smaller groups of proteins (clusters) which share more interactions among themselves and fewer with the remaining proteins of the network can lead to the discovery of protein complexes or functional modules (Spirin and Mirny, 2003). It is reasonable to assume that proteins appearing to be more closely connected must share a common function.

Until now, various computational approaches have been proposed in the academic world in the form of web-based or stand-alone software tools. Examples include applications such as NEAT (Brohee *et al.*, 2008) and jClust (Pavlopoulos *et al.*, 2009). However, most of these tools lack vital software application properties. In particular, we believe that the user should be able to execute various algorithms interactively. In addition, the ability to explore and navigate through PPI data visually is an important one. Interactive algorithm execution and visualisation of resulting PPI data make the interpretation of results easier for the scientist.

By using the Superclusteroid tool, the user can apply different clustering, visualisation and prediction methods in a continuous manner, which embraces user interaction. From the moment the user uploads the input data, all resulting files can be further manipulated in discrete stages, as the tool was specifically designed to bridge the compatibility gap amongst the various methods

Figure 1. Algorithm selection module of the Superclusteroid tool.

of each of the PPI data manipulation stages. Moreover, a variety of available PPI visualization modules are employed, in order to facilitate an intuitive result interpretation, where applicable.

Implementation

Design

Superclusteroid is a web-based application written in Perl and can be accessed using any internet browser able to execute a Java applet (note: although Java compatibility is not required for all operations, some of the visualisation tools do require the execution of a Java applet in the browser). It utilises already available clustering algorithms. As different algorithms provide different results (Brohee and van Helden, 2006; Li *et al.*, 2009), the user can choose among a set of widely used clustering algorithms to process the input data (Figure 1). These algorithms are: (i) MCL (Markov Cluster), an algorithm that computes the graph of random walks of an input graph, yielding a stochastic matrix (Enright *et al.*, 2002); (ii) Restricted Neighbourhood Search Clustering Algorithm (RNSC), a cost-based local search metaheuristic (King *et al.*, 2004); (iii) Highly Connected Subgraphs Algorithm (HCS), based on the detection of highly connected subgraphs (Hartuv and Shamir, 2000); (iv) SideS, a variation of HCS which uses a statistical model to express

the statistical significance of a cluster (Koyuturk *et al.*, 2007). It has to be noted that the additional algorithms (SideS and HCS) are not available on any other online tool, despite their efficiency on protein complex detection. The resulting files are tab-delimited data with two columns, one for the name of the cluster and one for the protein belonging to that cluster.

The above results can be automatically visualised or can be downloaded for later use. Additionally, the original network or other DOT files can be viewed by choosing the “visualize” tab on the home page, as it is shown in Figure 2. In either case, a java applet named “ZGRViewer” (Pietriga, 2005) is used to support the “fdp” and “twopi” [GraphViz/DOT tools](#) for spring model and radial layouts respectively. ZGRViewer is designed to handle large graphs, and offers a zoomable user interface (ZUI), which enables smooth zooming and easy navigation in the visualised structure. Furthermore, the user is able to visualise on a new tab of his/her browser a specific cluster. This dynamic visualisation module of Superclusteroid makes it easier for users to explore and analyse the clustering results, contrary to the static module of other web tools such as NEAT.

By choosing a specific protein, the user may continue with the analysis by implementing the Majority Vote Prediction Algorithm (MVPA) (Bu *et al.*, 2003) or the Hypergeometric Distribution Prediction Algorithm (HDP) (Enright *et al.*, 2002).

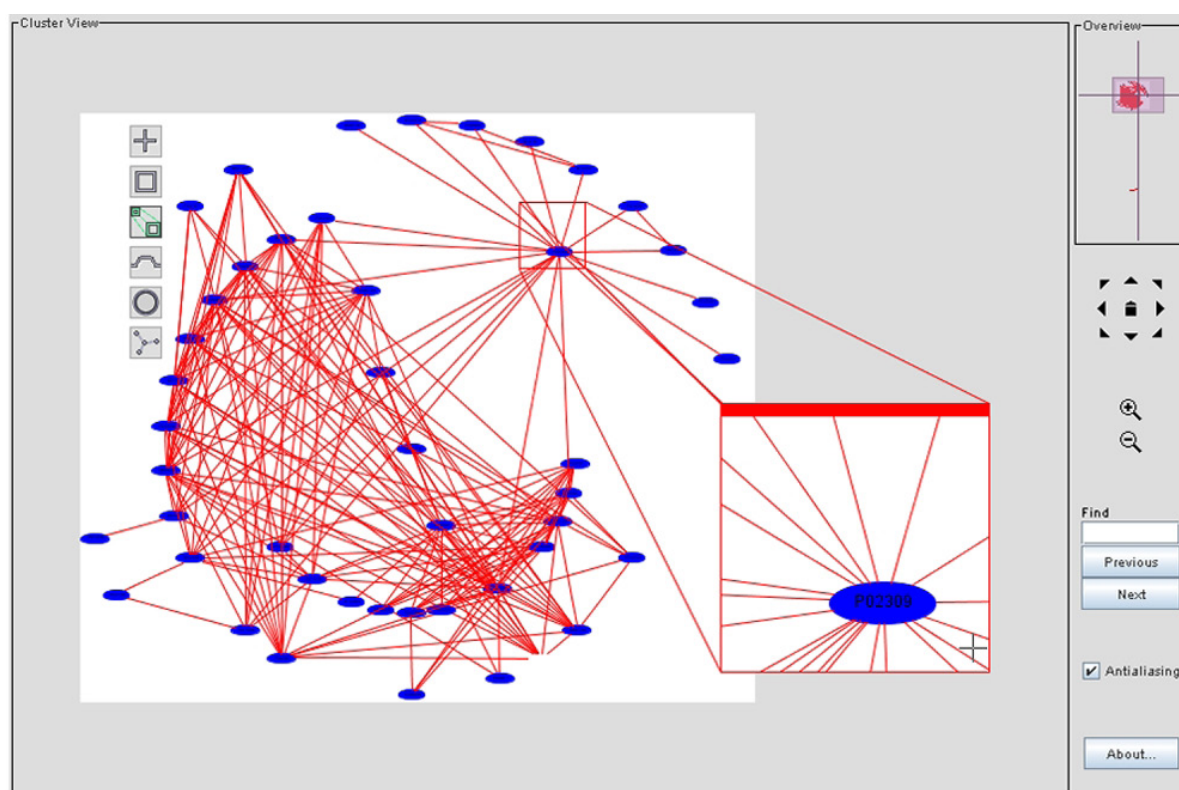


Figure 2. Visualisation module of Superclusteroid tool.

Both methods apply only for PPI data with Uniprot IDs and for the *S. cerevisiae* organism. The functional categories used are those provided in the FunCat database (Ruepp et al., 2004).

Input

The web tool can manipulate different input formats. Specifically, Superclusteroid supports tab-delimited text files, adjacency matrices in text files, DOT files-using the DOT network description languages and SIF files, a popular tab-delimited text file mostly used in Cytoscape (Shannon et al., 2003). The input file can be uploaded in an easy and quick manner in a user-friendly web page. Multiple identical PPIs are removed from further analysis. More information about the input data format is available at Superclusteroid help pages.

For the purpose of presenting Superclusteroid, the Gavin 2006 dataset (Gavin et al., 2006) is used for all four algorithms available and produces the required clustering results using the default parameters. In order to prove the tool's ability to predict protein complexes, the four different clustering results are compared with the

recorded protein complexes stored in the MIPS database concerning the *S. cerevisiae* organism (Mewes et al., 2002). The recorded complexes of the MIPS database are used as a golden standard in order to compare the results of the each time applied algorithm (Brohee and van Helden, 2006; Li et al., 2009).

Table 1. The MVPA function category scores of the protein P38334.

#	Category	Score
1	Cellular transport, transport facilities and transport routes	9
2	Metabolism	2
3	Biogenesis of Cellular Components	1
4	Cell Type Differentiation	1
5	Cell Cycle and DNA processing	1
6	Energy	1

Table 2. The HDPA function category scores of the protein P38334

#	Category	Score
1	Cellular transport, transport facilities and transport routes	1.10 E-05
2	Energy	0.504003
3	Cell Type Differentiation	0.647883
4	Metabolism	0.761164
5	Biogenesis of Cellular Components	0.902691
6	Cell Cycle and DNA processing	0.95261

Results

In order to prove the efficiency of Superclusteroid compared to other similar clustering tools, we performed experiments using the Gavin 2006 dataset (Gavin *et al.*, 2006). This dataset consists of 1,430 proteins and 6,531 interactions which derived from Tandem Affinity Purification method (Puig *et al.*, 2001) and Mass Spectrometry (Ho *et al.*, 2002).

We chose to use the MCL algorithm which according to (Brohee and van Helden, 2006) and (Li *et al.*, 2009), is one of the best clustering algorithms. The initial dataset was divided into 188 clusters, where each of these can be visualised and manipulated independently. Then we chose randomly a protein, the one called P38334, and we tried to determine its functionality by using the corresponding Superclusteroid module. Tables 1 and 2 show the results of the MVPA and the HDPA algorithms.

In both cases, the function category "Cellular transport, transport facilities and transport routes", according to the FUNCAT database, is the most likely for the protein P38334.

By using the UniProt database (Magrane and Consortium, 2011), it can be seen that P38334 is part of the TRAPP complex (Sacher *et al.*, 1998), which according to Gene Ontology data (Barrell *et al.*, 2009), is a large complex on the cis-Golgi that mediates vesicle docking and fusion. It is divided into two parts: TRAPP I, which is a multisubunit complex that consists of seven subunits, and TRAPP II, which has three additional subunits and that functions as a tether at latter stages of the transport pathway. Therefore, the Superclusteroid successfully predicted the functionality of the P38334 protein, which is a service that is not provided by other similar clustering tools.

Conclusion

Our results prove that Superclusteroid is capable of predicting protein complexes in an easy-to-use way. Additionally, data formats can be easily manipulated and clustering results can be cross-referenced as the tool provides four different clustering algorithms. Superclusteroid also detects complexes that do not match any confirmed complex in MIPS database. As we cluster the complete interactome, of which the confirmed complexes provide only partial coverage, we speculate that complexes detected by our method could match yet unknown or unconfirmed protein complexes. However, it must be emphasised that protein complexes are not the only ones that can be detected. As explained earlier, the clustering algorithms provide protein groups that are more "connected" among themselves. This statistical significance does not apply specifically to protein complexes, but it is also applicable to functional modules. This term is used for proteins that participate in a common cellular process while binding each other at a different time and place (Spirin and Mirny, 2003).

To sum up, Superclusteroid: (i) uploads and manipulates input of PPI data; (ii) performs clustering on PPI data using four different algorithms; (iii) visualises PPI networks and clustering results; (iv) predicts protein function. It can be used for the prediction of protein complexes in a user-friendly way. Superclusteroid also provides a help page that contains explicit instructions describing its services and a comprehensive list of the web services available, along with their description and the access URL for each of them. Additionally, the web tool provides demo data to help the user to understand its functionality.

The tool is implemented in the GNU/Linux environment and is written in Perl¹. In addition to the website, web services utilising the SOAP protocol² are also available in order to design workflows and integrate them with other available resources.

Acknowledgements

We would like to thank the University Biotechnology Center of Oslo for hosting the Superclusteroid web tool. We also would like to thank Erik Bongcam-Rudloff for organising the joint EMBnet-EMBRACE workshop on creating web services for Bioinformatics (Uppsala, Sweden, 2008).

References

1. Barrell D, Dimmer E, Huntley R P, Binns D, O'Donovan C, Apweiler R (2009) The GOA database in 2009--an integrated Gene Ontology Annotation resource. *Nucleic Acids Res* 37, (Database issue) D396-403.
2. Brohee S, Faust K, Lima-Mendez G, Sand O, Janky R, Vanderstocken G, Deville Y, van Helden J (2008) NeAT: a toolbox for the analysis of biological networks, clusters, classes and pathways. *Nucleic Acids Res* 36, W444-51.
3. Brohee S, van Helden J (2006) Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics* 7, 488.
4. Bu D, Zhao Y, Cai L, Xue H, Zhu X, Lu H, Zhang J, Sun S, Ling L, Zhang N, Li G, Chen R (2003) Topological structure analysis of the protein-protein interaction network in budding yeast. *Nucleic Acids Res* 31, 2443-50.
5. Enright A J, Van Dongen S, Ouzounis C A (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res* 30, 1575-84.
6. Gavin A C, Aloy P, Grandi P, Krause R, Boesche M, Marzioch M, Rau C, Jensen L J, Bastuck S, Dumpelfeld B, Edelmann A, Heurtier M A, Hoffman V, Hoefert C, Rudi K, Hudak M, Michon A M, Schelder M, Schirle M, Remor M, Rudi T, Hooper S, Bauer A, Bouwmeester T, Casari G, Drewes G, Neubauer G, Rick J M, Kuster B, Bork P, Russell R B, Superti-Furga G (2006) Proteome survey reveals modularity of the yeast cell machinery. *Nature* 440, 631-6.
7. Gavin A C, Bosche M, Krause R, Grandi P, Marzioch M, Bauer A, Schultz J, Rick J M, Michon A M, Cruciat C M, Remor M, Hofert C, Schelder M, Brajenovic M, Ruffner H, Merino A, Klein K, Hudak M, Dickson D, Rudi T, Gnau V, Bauch A, Bastuck S, Huhse B, Leutwein C, Heurtier M A, Copley R R, Edelmann A, Querfurth E, Rybin V, Drewes G, Raida M, Bouwmeester T, Bork P, Seraphin B, Kuster B, Neubauer G, Superti-Furga G (2002) Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* 415, 141-7.
8. Hartuv E, Shamir R (2000) A clustering algorithm based on graph connectivity. *Information Processing Letters* 76, 175-181.
9. Ho Y, Gruhler A, Heilbut A, Bader G D, Moore L, Adams S L, Millar A, Taylor P, Bennett K, Boutilier K, Yang L, Wolting C, Donaldson I, Schandorff S, Shewnarane J, Vo M, Taggart J, Goudreau M, Muskat B, Alfano C, Dewar D, Lin Z, Michalickova K, Willems A R, Sassi H, Nielsen P A, Rasmussen K J, Andersen J R, Johansen L E, Hansen L H, Jespersen H, Podtelejnikov A, Nielsen E, Crawford J, Poulsen V, Sorensen B D, Matthiesen J, Hendrickson R C, Gleeson F, Pawson T, Moran M F, Durocher D, Mann M, Hogue C W, Figeys D, Tyers M (2002) Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry. *Nature* 415, 180-3.
10. Ito T, Chiba T, Ozawa R, Yoshida M, Hattori M, Sakaki Y (2001) A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proceedings of the National Academy of Science* 98, 4569-4574.
11. King A D, Przulj N, Jurisica I (2004) Protein complex prediction via cost-based clustering. *Bioinformatics* 20, 3013-20.
12. Koyuturk M, Szpankowski W, Grama A (2007) Assessing significance of connectivity and conservation in protein interaction networks. *J Comput Biol* 14, 747-64.
13. Li X, Wu M, Kwok C K, Ng S K (2009) Computational approaches for detecting protein complexes from protein interaction networks: a survey. *BMC Genomics* 11 (Suppl 1), S3.
14. Magrane M, Consortium U (2011) UniProt Knowledgebase: a hub of integrated protein data. *Database (Oxford)* bar009.
15. Mewes H W, Frishman D, Guldener U, Mannhaupt G, Mayer K, Mokrejs M, Morgenstern B, Munsterkotter M, Rudd S, Weil B (2002) MIPS: a database for genomes and protein sequences. *Nucleic Acids Res* 30, 31-4.
16. Pavlopoulos G A, Moschopoulos C N, Hooper S D, Schneider R, Kossida S (2009) jClust: a clustering and visualization toolbox. *Bioinformatics* 25, 1994-6.
17. Pietriga E (2005) A Toolkit for Addressing HCI Issues in Visual Language Environments. *EEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'05)*, 145-152.
18. Puig O, Caspary F, Rigaut G, Rutz B, Bouveret E, Bragado-Nilsson E, Wilm M, Seraphin B (2001) The tandem affinity purification (TAP) method: a gen-

¹ <http://www.perl.com/>

² <http://www.w3.org/TR/soap/>

- eral procedure of protein complex purification. *Methods* **24**, 218-29.
19. Ruepp A, Zollner A, Maier D, Albermann K, Hani J, Mokrejs M, Tetko I, Guldener U, Mannhaupt G, Munsterkotter M, Mewes H W (2004) The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucleic Acids Res* **32**, 5539-45.
20. Sacher M, Jiang Y, Barrowman J, Scarpa A, Burston J, Zhang L, Schieltz D, Yates J R, 3rd, Abeliovich H, Ferro-Novick S (1998) TRAPP, a highly conserved novel complex on the cis-Golgi that mediates vesicle docking and fusion. *EMBO J* **17**, 2494-503.
21. Shannon P, Markiel A, Ozier O, Baliga N S, Wang J T, Ramage D, Amin N, Schwikowski B, Ideker T (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res* **13**, 2498-504.
22. Spirin V, Mirny L A (2003) Protein complexes and functional modules in molecular networks. *Proc Natl Acad Sci U S A* **100**, 12123-8.
23. Stoll D, Templin M F, Bachmann J, Joos T O (2005) Protein microarrays: applications and future challenges. *Curr Opin Drug Discov Devel* **8**, 239-52.
24. Willats W G (2002) Phage display: practicalities and prospects. *Plant Mol Biol* **50**, 837-54.

Taxonomic Assignment in Metagenomics with TANGO



**Daniel Alonso-Aleman¹, José C. Clemente²,
Jesper Jansson³, Gabriel Valiente⁴**

¹Algorithms, Bioinformatics, Complexity and Formal Methods Research Group, Technical University of Catalonia, E-08034 Barcelona, Spain

²Department of Chemistry and Biochemistry, University of Colorado, Boulder, CO, USA

³Ochanomizu University, 2-1-1 Otsuka, Bunkyo-ku, Tokyo 112-8610, Japan

⁴Algorithms, Bioinformatics, Complexity and Formal Methods, Research Group, Technical University of Catalonia, E-08034 Barcelona, Spain

Corresponding author: valiente@lsi.upc.edu

Abstract

One of the main computational challenges facing metagenomic analysis is the taxonomic identification of short DNA fragments. The combination of sequence alignment methods with taxonomic assignment based on consensus can provide an accurate estimate of the microbial diversity in a sample. In this note, we show how recent improvements to these consensus methods, as implemented in the latest release of the TANGO tool, can provide an improved estimate of diversity in simulated datasets.

Introduction

The diversity and richness of microbial populations can be characterised by several ecological indices, calculated by either grouping similar sequence reads into operational taxonomic units, or assigning them to the most similar taxa in a given taxonomy. While the former is useful for the study of unknown microbial communities, the latter is best suited when sequences and taxonomies of related species are already known.

The usual protocol for taxonomic assignment involves aligning the sequence reads to a set of reference sequences and, then, resolving any ambiguities (that is, a sequence being equally similar to more than one reference sequence) by assigning to a consensus sequence, such as the lowest common ancestor (LCA) of all the candidate sequences in a given taxonomy (Huson *et al.*, 2007; Kunin *et al.*, 2008; Liu *et al.*, 2008). Sequence composition-based methods have also been used in taxonomic assignment (Diaz *et al.*, 2009; McHardy *et al.*, 2007; Wang *et al.*, 2007).

Previous work on taxonomic assignment based on alignment has focused either on sequence reads of the 16S ribosomal RNA gene (Clemente *et al.*, 2010, 2011; Ribeca and Valiente, 2011), or on whole metagenomic shotgun sequence reads (Gerlach *et al.*, 2009; Krause *et al.*, 2008). In this note, we show for the latter that recent improvements to consensus methods, as implemented in the latest release of the TANGO tool (Clemente *et al.*, 2011), bring about an accurate estimate of the actual taxonomic diversity in a metagenomic data-set.

In the improved consensus method, ambiguous sequence reads are assigned to consensus sequences at a lower taxonomic rank than the LCA of the candidate reference sequences (increased specificity), at the expense of discarding some candidate reference sequences (reduced sensitivity). This is done by optimising the combined precision and recall (F-measure) of the taxonomic assignment (Clemente *et al.*, 2010, 2011).

Metagenomic data-set

The complexity of the signal obtained when sequencing metagenomic data makes it necessary to take a standardised data-set as the basis for analysis (Ribeca and Valiente, 2011). We have chosen the metagenomic data-set of Mavromatis *et al.* (2007), which was designed with the goal of simulating microbial communities of varying complexity: low-complexity communities, with one dominant population (simLC), as seen in bioreactor communities (García Martín *et al.*, 2006; Strous *et al.*, 2006); medium-complexity communities, with more than one dominant population flanked by low-abundance populations (simMC), as seen in acid mine drainage biofilm (Tyson *et al.*, 2004).

Table 1. Phylogenetic distribution of the 113 microbial genomes.

Domain	Phylum	Class	Genomes
Bacteria	Actinobacteria	Actinobacteria	9
	Bacteroidetes	Cytophagia	1
	Chlorobi	Chlorobia	7
	Chloroflexi	Chloroflexi	1
	Cyanobacteria	Cyanobacteria	6
	Deinococcus-Thermus	Deinococci	1
	Firmicutes	Bacilli	13
		Clostridia	8
	Proteobacteria	Alphaproteobacteria	17
		Betaproteobacteria	13
		Gammaproteobacteria	25
		Deltaproteobacteria	6
		Epsilonproteobacteria	1
		unclassified Proteobacteria	1
Archaea	Euryarchaeota	Methanomicrobia	3
		Thermoplasmata	1

and symbiotic microbes from eukaryotes (Woyke *et al.*, 2006); and high-complexity communities, with no dominant population (simHC), as seen in agricultural soil (Tringe *et al.*, 2005).

The Mavromatis *et al.* data-set was built by combining Sanger sequence reads selected at random from 113 microbial genomes. The phylogenetic composition of the metagenomic data-set, summarised in Table 1, shows a high abundance of Proteobacteria, Actinobacteria, and Firmicutes, as usual in most metagenomic samples (Gabor *et al.*, 2004; Manichanh *et al.*, 2008).

The distribution of sequence reads in the metagenomic data-set, summarised in Table 2, shows a low-complexity microbial community, with one dominant population (28,861 sequence reads from *Rhodopseudomonas palustris* HaA2); a medium-complexity microbial community, with three dominant populations (22,956 sequence reads from *Bradyrhizobium* sp. BTAi1, 16,577 sequence reads from *Rhodopseudomonas palustris* BisB5, and 10,484 sequence reads from *Xylella fastidiosa* Dixon) flanked by low-abundance populations; and a high-complexity microbial community, with no dominant population.

Table 2. Distribution of sequence reads in the metagenomic data-set.

	simLC	simMC	simHC
Most abundant	28,861	22,956	2,384
2 nd abundant	9,277	16,577	2,248
3 rd abundant	5,168	10,484	2,191
4 th abundant	1,149	6,107	2,127
5 th abundant	1,109	4,868	2,083
6 th abundant	1,074	1,146	2,051
Rest	50,857	52,319	103,687

Aligning sequence reads

The first step in the taxonomic analysis of a metagenomic data-set involves aligning the sequence reads to a database of known sequences from a large set of different organisms. Traditional alignment tools, such as BLAST (Altschul *et al.*, 1990) or BLAT (Kent, 2002), do not scale up to align millions or billions of sequence reads to a large reference genome (Horner *et al.*, 2010; Ribeca and Valiente, 2011; Trapnell and Salzberg, 2009). Microbial genomes are much shorter, though, making these tools appropriate for the alignment of sequence reads from envi-

Table 3: Ambiguous sequence reads in the metagenomic data-set.

Data-set	No hit	One hit	Ambiguous	Total
simLC	59	22,956	2,384	97,495
simMC	76	16,577	2,248	114,457
simHC	100	10,484	2,191	116,771

ronmental samples. Nevertheless, more efficient tools are available for the alignment of short and long sequence reads obtained using high-throughput sequencing technologies, including BWA (Li and Durbin, 2009), BWA/SW (Li and Durbin, 2010), and GEM (Ribeca, 2009).

We have used BLAST to align the 328,723 sequence reads to the 113 microbial genomes. Notice that a larger database is often used when the target sequences are not known beforehand. Ambiguities arise when a sequence read is aligned with more than one target sequence, and we have taken as candidate alignments all those sequences with the same E-value as the top BLAST hit. As shown in Table 3, ambiguous sequence reads represent about 20% of the metagenomic data-set. Sequence reads with no hit in the database of microbial genomes are the result of sequencing errors.

Assigning sequence reads

Once the sequence reads have been aligned to reference sequences, the second step in the taxonomic analysis of a metagenomic data-set involves resolving ambiguities by mapping those reads with more than one possible assignment to species at the closest possible taxonomic rank. We have chosen as taxonomic reference the NCBI taxonomy (Sayers *et al.*, 2009) for the 113 sampled microbial genomes. Again, no-

tice that a larger taxonomy is often used when the target sequences are not known beforehand. Alternative taxonomies for microbial genomes include ARB-SILVA (Pruesse *et al.*, 2007), Greengenes (DeSantis *et al.*, 2006), RDP (Cole *et al.*, 2009), and TOBA (Garrity *et al.*, 2007).

We have used TANGO to assign the 328,723 sequence reads to the 113 microbial genomes at the closest possible taxonomic rank. As shown in Table 4, the optimal consensus method, F-measure-based assignment, resulted in assignments at a lower taxonomic rank than the classical consensus method, LCA-based assignment (Huson *et al.*, 2007).

Taxonomic diversity

Once the sequence reads have been assigned a taxonomy, the third and final step in the taxonomic analysis of a metagenomic data-set involves describing the diversity and richness of the sampled microbial population by means of ecological indices. Some widely accepted notions in ecology are those of α -diversity (species diversity within an ecosystem), β -diversity (change in species diversity within an ecosystem), and ω -diversity (phylogenetic difference between species in an ecosystem) (Faith, 1992; Whittaker, 1972). Among the latter, we have chosen the Clarke-Warwick taxonomic diversity index (Clarke and Warwick, 1998), which measures the

Table 4: Taxonomic distribution of the metagenomic data-set using consensus (LCA, top) and optimal (F-measure, bottom) taxonomic assignment.

Data-set	Taxonomic rank					
	Domain	Phylum	Class	Order	Family	Genus
simLC	126	104	134	56	2,785	5,295
simMC	194	176	174	101	2,784	5,219
simHC	272	219	230	111	822	11,164
simLC		1	65	46	1,236	3,241
simMC		10	90	104	1,179	3,191
simHC		12	145	77	414	6,847

Table 5: Taxonomic diversity (Clarke-Warwick index) of the metagenomic data-set for consensus (LCA) and optimal (F-measure) taxonomic assignment, together with the actual taxonomic diversity.

Data-set	Taxonomic diversity	
	LCA	F-measure
simLC	3.8193	4.5798
simMC	4.1485	4.7993
simHC	4.9433	5.7422

average distance in the taxonomic reference between the sampled species.

As shown in Table 5, the closer the measured taxonomic diversity in the metagenomic data-set is to the actual taxonomic diversity in the sampled population, the more accurate the assignment is: that is, when classical consensus (LCA) is replaced by the optimal consensus (F-measure) method.

Conclusion

The combination of sequence alignment methods with taxonomic assignment based on consensus provides an accurate estimate for the composition of a sample of sequence reads of the 16S ribosomal RNA gene. We have shown that for sequence reads of whole microbial genomes, recent improvements to consensus methods also bring about an accurate estimate of the microbial diversity in a metagenomic sample.

Acknowledgements

DA was supported by the Ministry of Economy and Knowledge of the Government of Catalonia and the European Social Fund. JJ was supported by the Special Coordination Funds for Promoting Science and Technology, Japan.

References

1. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. *J Mol Biol* **215**, 403-410. doi:10.1016/S0022-2836(05)80360-2
2. Clarke KR, Warwick RM (1998) A taxonomic distinctness index and its statistical properties. *J Appl Ecol* **35**, 523-531. doi:10.1046/j.1365-2664.1998.3540523.x
3. Clemente JC, Jansson J, Valiente G (2010) Accurate taxonomic assignment of short pyrosequencing reads. *Pacific Symp Biocomput* **15**, 3-9. doi:10.1142/9789814295291_0002
4. Clemente JC, Jansson J, Valiente G (2011) Flexible taxonomic assignment of ambiguous sequencing reads. *BMC Bioinformatics* **12**, 8. doi:10.1186/1471-2105-12-8
5. Cole JR, Wang Q, Cardenas E, Fish J, Chai B et al. (2009) The Ribosomal Database Project: improved alignments and new tools for rRNA analysis. *Nucleic Acids Res* **37**, D141-D145. doi:10.1093/nar/gkn879
6. DeSantis TZ, Hugenholtz P, Larsen N, Rojas M, Brodie EL et al. (2006) Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl Environ Microbiol* **72**, 5069-5072. doi:10.1128/AEM.03006-05
7. Diaz NN, Krause L, Goesmann A, Niehaus K, Nattkemper TW (2009) TACO: taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach. *BMC Bioinformatics* **10**, 56.
8. Faith DP (1992) Conservation evaluation and phylogenetic diversity. *Biol Conserv* **61**, 1-10. doi:10.1016/0006-3207(92)91201-3
9. Gabor EM, Alkema WBL, Janssen DB (2004) Quantifying the accessibility of the metagenome by random expression cloning techniques. *Environ Microbiol* **6**, 879-886. doi:10.1111/j.1462-2920.2004.00640.x
10. García Martín H, Ivanova N, Kunin V, Warnecke F, Barry KW et al. (2006) Metagenomic analysis of two enhanced biological phosphorus removal (EBPR) sludge communities. *Nat Biotechnol* **24**, 1263-1269. doi:10.1038/nbt1247
11. Garrity GM, Lilburn TG, Cole JR, Harrison SH, Euzéby J et al. (2007) The taxonomic outline of bacteria and archaea. TOBA release 7.7. Michigan State University Board of Trustees, <http://www.taxonomic-outline.org/>.
12. Gerlach W, Jünemann S, Tille F, Goesmann A, Stoye J. (2009) WebCARMA: a web application for the functional and taxonomic classification of unassembled metagenomic reads. *BMC Bioinformatics* **10**, 430. doi:10.1186/1471-2105-10-430
13. Horner DS, Pavesi G, Castrignanò T, De Meo PD, Liuni S et al. (2010) Bioinformatics approaches for genomics and post genomics applications of nextgeneration sequencing. *Brief Bioinform* **11**, 181-197. doi:10.1093/bib/bbp046
14. Huson DH, Auch AF, Qi J, Schuster SC (2007) MEGAN analysis of metagenomic data. *Genome Res* **17**, 377-386. doi:10.1101/gr.5969107
15. Kent WJ (2002) BLAT—The BLAST-like alignment tool. *Genome Res* **12**, 656-664. doi:10.1101/gr.229202
16. Krause L, Diaz NN, Goesmann A, Kelley S, Nattkemper TW et al. (2008) Phylogenetic classification of short environmental DNA fragments.

- Nucleic Acids Res* **36**, 2230-2239. doi:10.1093/nar/gkn038
17. Kunin V, Copeland A, Lapidus A, Mavromatis K, Hugenholtz P (2008) A bioinformatician's guide to metagenomics. *Microbiol Mol Biol Rev* **72**, 557-578. doi:10.1128/MMBR.00009-08
18. Li H, Durbin R (2009) Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* **25**, 1754-1760. doi:10.1093/bioinformatics/btp324
19. Li H, Durbin R (2010) Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics* **26**, 589-595. doi:10.1093/bioinformatics/btp698
20. Liu Z, DeSantis TZ, Andersen GL, Knight R (2008) Accurate taxonomy assignments from 16S rRNA sequences produced by highly parallel pyrosequencers. *Nucleic Acids Res* **36**, e120. doi:10.1093/nar/gkn491
21. Manichanh C, Chapple CE, Frangeul L, Gloux K, Guigo R *et al.* (2008) A comparison of random sequence reads versus 16S rDNA sequences for estimating the biodiversity of a metagenomic library. *Nucleic Acids Res* **36**, 5180-5188. doi:10.1093/nar/gkn496
22. Mavromatis K, Ivanova N, Barry K, Shapiro H, Goltsman E *et al.* (2007) Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. *Nat Methods* **4**, 495-500. doi:10.1038/nmeth1043
23. McHardy AC, García Martín H, Tsirigos A, Hugenholtz P, Rigoutsos I (2007) Accurate phylogenetic classification of variable-length DNA fragments. *Nat Methods* **4**, 63-72. doi:10.1038/nmeth976
24. Pruesse E, Quast C, Knittel K, Fuchs BM, Ludwig W *et al.* (2007) SILVA: A comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB. *Nucleic Acids Res* **35**, 7188-7196. doi:10.1093/nar/gkm864
25. Ribeca P (2009) GEM: Genomic multi-tool. <http://gemlibrary.sourceforge.net/>.
26. Ribeca P, Valiente G (2011) Computational challenges of sequence classification in microbiomic data. *Brief Bioinform.* In press. doi:10.1093/bib/bbr019
27. Sayers EW, Barrett T, Benson DA, Bolton E, Bryant SH *et al.* (2011). Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res* **39**, D38-D51. doi:10.1093/nar/gka1172
28. Strous M, Pelletier E, Mangenot S, Rattei T, Lehner A *et al.* (2006) Deciphering the evolution and metabolism of an anammox bacterium from a community genome. *Nature* **440**, 790-794. doi:10.1038/nature04647
29. Trapnell C, Salzberg SL (2009) How to map billions of short reads onto genomes. *Nat Biotechnol* **27**, 455-458. doi:10.1038/nbt0509-455
30. Tringe SG, von Mering C, Kobayashi A, Salamov AA, Chen K *et al.* (2005) Comparative metagenomics of microbial communities. *Science* **308**, 554-557. doi:10.1126/science.1107851
31. Tyson GW, Chapman J, Hugenholtz P, Allen EE, Ram RJ *et al.* (2004) Community structure and metabolism through reconstruction of microbial genomes from the environment. *Nature* **424**, 37-43.
32. Wang Q, Garrity GM, Tiedje JM, Cole JR (2007) Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. *Appl Environ Microbiol* **73**, 5261-5267. doi:10.1128/AEM.00062-07
33. Whittaker RH (1972) Evolution and measurement of species diversity. *Taxon* **21**, 213-251.
34. Woyke T, Teeling H, Ivanova NN, Huntemann M, Richter M *et al.* (2006) Symbiosis insights through metagenomic analysis of a microbial consortium. *Nature* **443**, 950-955. doi:10.1038/nature05192

Using the Grid to run population dynamics simulations



José R. Valverde

EMBnet/CNB, Centro Nacional de Biotecnología, CSIC.
C/Darwin, 3. 28049 Madrid. Spain

Abstract

Analysis of population evolutionary dynamics using realistic models is a challenging task requiring access to huge resources. Estimates for simple models of population growth under different mutation and selection conditions yield running times of Central Processing Unit (CPU) years. As mutations are stochastic events, experiments can be split into many separate jobs, reducing to a large Monte Carlo-like problem that is embarrassingly parallel and thus maps perfectly on the Grid.

We have been able to run simulations with realistic population sizes (up to 1,000,000 individuals) and growth cycles using the Grid with a ~190x efficiency gain, thus reducing execution time from years to a few days. This speed-up allows us to accelerate the simulation cycle, and work on data analysis and additional model refinements with minimal delays and effort.

We have taken measures at various steps in the process to study the efficiency gains obtained. While our simple approach may arguably be far from achieving optimum efficiency, we were able to achieve significant gains. Here, we analyse Grid efficiency and discuss which benefits can be realistically expected with the current technology; we also provide useful advice for future Grid developers.

All the tools described are available under GNU's Public License (GPL) from http://ahriman.cnb.csic.es/sbg/tiki-download_file.php?fileid=16

Introduction

Building realistic population simulations is a typical embarrassingly parallel large-scale computation. This kind of problem maps naturally to massively distributed architectures, like the [EGEE Grid](#)¹ (Enabling Grids for E-science in Europe). Solving this instance therefore provides solid ground both for solving other similar tasks and for testing the adequacy of current technology.

Our main interest was to study the selection processes taking place in bacteria with different mutation rates. The problem of itself is interesting for many reasons: from a theoretical point of view, it is a simplified model of the evolution of more complex organisms and ecosystems; but

it also has relevant practical implications to further our understanding of population dynamics, evolution and mutation rates, and to understand the development of interesting traits, like bacterial resistance to antibiotics.

There is a relevant interest in solving, or at least understanding, the problem in detail; however, while growing a bacterial population in the laboratory is cheap routine work, analysing the evolution and selection of gene mutations experimentally is not so simple, as it would require genotyping of representative samples of bacterial populations and assessment of the impact of each selected genotype on the viability of its carrier (Sniegowski *et al.*, 1997).

Because experimental validation is inconvenient, it is desirable to model *in silico* what would happen in the test tube. The main problem now is being able to produce realistic simulations: as cell division is an exponential process, we soon find ourselves modelling large numbers of specimens, whose mutation events must be tracked, and we need to collect statistically significant data.

Running these simulations has largely been constrained by technological limitations, resulting in reductionist models that (despite their shortcomings) have harvested useful insights on the problem (Wilke *et al.*, 2001; Lenski *et al.*, 1999; Adami *et al.*, 2000; Taddei *et al.*, 1997; Johnson, 1999). Despite Moore's law, running a realistic simulation easily results in very long computation times, limiting its usefulness. More specifically, our estimates for the simulation we wanted to run were in the order of years of CPU time.

Our simulations use a Monte Carlo method: we repeat a basic experiment enough times to collect statistically sound results. Additionally, because each simulation experiment is independent from all others, by simply using a different seed, our approach may be generalised to any embarrassingly parallel system with a large number of non-communicating tasks.

Finally, because simulated population growth is affected by mutation rates and the effect of random mutations on viability, varying initial conditions have a large impact on population size during the simulation, resulting in large variability of simulation run times, posing additional challenges and making ours a problem of more generic interest.

¹ www.eu-egee.org

This paper deals with the implementation details of these simulations on the Grid. Our population dynamics simulations are still being further refined, although preliminary results from the analysis involving various combinations of different mutator phenotypes, selection coefficients and mutation rates led to two main scenarios, demanding more extensive analysis; these were presented as part of the 2007 Workshops, Current Trends in Biomedicine series, *"Stress, stress responses and mechanisms of evolvability"* at the Universidad Internacional de Andalucia, Baeza, Spain, 2007, and will be fully discussed once the analysis and experimental verification have been completed in a separate publication.

Methods

Simulation code

The population dynamics simulation was based on in-house code written in Fortran95, requiring no additional libraries or dependencies. The long run-times required for a realistic simulation necessitated the problem to be split into sub-problems suitable for running on the EGEE Grid. All programs were compiled statically using the Gfortran compiler to avoid library dependencies on remote hosts.

Each experiment tests a set of constraints under a large variety of initial parameters (up to 1,000), executing a sensible number of simulated culture cycles (up to 100). The initial model simulated laboratory conditions, using in each culture cycle an inoculate of individuals with several genes, taken from a previous culture, that would undergo many replication, mutation, competition and selection events until a sensibly large colony size (usually of the order of a million individuals), or number of replication events, was reached.

Output of each simulation run was used to further refine and optimise the initial model, making it more meaningful. This refinement process is still an ongoing concern.

Owing to the large variation of constraints, run-times also show large variation, as may be expected: a population suffering more deleterious mutations grows less, its reduced number of individuals resulting in lesser simulation resource and time requirements.

Grid parallelisation

The simulation was conducted to mimic many *in vivo* experiments under controlled starting conditions. Because mutation is a stochastic process, we could split work into separate runs using different random seeds. To manage jobs, we developed tools that have been progressively refined to adapt to various issues and shortcomings.

The job-management scripts were developed as shell scripts, and can be coarsely classified into three categories: a set of scripts to generate the large number of jobs required; a set of generic scripts to launch jobs, monitor their status and collect results; and a set to process the results into manageable statistics.

Job management was designed as a set of generic scripts that can be used for any kind of non-specific job: the system expects all jobs for an experiment to be collected in a single directory, with each job being stored in a separate, self-contained sub-directory with all data and software needed for the computation. Submission works by traversing all job sub-directories, making links to generic Job Definition Language (JDL) and execution script files, and independently sending each job to an appropriate resource broker. Failure recovery involves traversal of the job sub-directories to search for aborted, failed or silently dead jobs and resubmitting them up

```
Type = "job";
JobType = "normal";
VirtualOrganisation = "biomed";
Executable = "job.sh";
StdOutput = "std.out";
StdError = "std.err";
InputSandbox = {"job.sh", "program", "input"};
OutputSandbox = {"std.out", "std.err", "result.dat"};
```

Figure 1. A typical job.jdl file may be as simple or complex as needed.


```
#!/bin/bash
#
chmod 755 program
./program < input
```

Figure 2. A typical job.sh script.

to a maximum number of tries. Data collection checks job status for successful termination and retrieves the output from the Grid into the job directory. The whole process is managed from a higher-level script that controls the timing of submission, failure recovery and output retrieval until all jobs have successfully finished.

With generic job management in place, it is now easy to automate generation of the large numbers of jobs required: only a generic execution script and JDL file need to be written, and copied by the submission system to the job sub-directory; and a simple script or shell loop-command are also needed to create the job sub-directories, copy (or better, hard link to save space) any common files, and generate any specific files depending on job parameters (Figures 1, 2 and 3).

Data collection and analysis were similarly performed by a set of scripts or shell commands: all that was needed was a loop traversing every job sub-directory and parsing output to extract relevant information.

Execution of data collection

In order to assess the impact of Grid architecture on the efficiency gains obtained, we inserted in our code specific instructions to collect timing data at various key steps, so that we could

measure the time invested at each step and investigate its influence on overall performance. The steps chosen were as follows: start and end of job submission (s_o , s_i); start and end of job execution (e_o , e_i) at the Working Node (WN); detection of job termination/start of result retrieval, and end of result retrieval (r_o , r_i).

Collecting times on the Grid requires additional care, as different steps will take place in different time zones. We took advantage of the fact that the Grid has a universal time and clock synchronisation, and measured time in Universal Coordinated Time (UTC) to avoid local offsets.

Another issue worth considering is the underlying WN architecture, as different machines may lead to different execution speeds. While this is intuitively true, we didn't consider it because it must be coupled to an unknown factor: a given WN may be simultaneously running more than one job at different priorities, hence, perhaps counter-intuitively, a loaded high-speed computer might perform worse than an old slower machine. Because there is no way to know which other tasks a given node is executing, at what priority, or for how long they overlap our job, this issue was not dealt with.

As our programs were compiled only for a 32-bit architecture, we also did not examine architecture-specific (64- vs. 32-bit) differences.

```
for i in {10..50..10}; do
  for j in {1..20}; do
    job=$i-`printf %02d $j` ;
    mkdir $job
    cd $job
    ln ../../exe/program .
    echo "$i $j" > input
    cd ..
  done
done
done
```

Figure 3. A typical job-generation command.

Results

Choice of computing system

From preliminary measures, we expected full experiment simulations to need from one to several years of CPU time for each experiment. This prompted us to seek other alternatives. Our two main options were the *Marenostrum* massively parallel supercomputer and the EGEE Grid. We opted for the Grid owing to its simplicity and immediate availability.

The problem reduces to a very large Monte Carlo simulation of mutation events on a dynamically growing population. We could further simplify the simulation by dividing it into separate growth cycles, much like one would do in laboratory practice.

Running one simulation on the Grid

We first tried to shift the parallel/serial balance towards computation by trying to fit all growth cycles for a given parameter-set in one process. One experiment would therefore require as many jobs as different initial conditions (hundreds). Each job was submitted and monitored separately.

This results in many sleeping processes waiting on the system for their monitored jobs to terminate, to the detriment of other concurrent users. Moreover, we observed that a discouragingly high number of jobs (~40%) aborted on execution. Investigation showed that *many sites maintain short-lived batch queues with execution times of 72 hours or less*. Because our problem could be further split with little extra work, we therefore decided to generate a larger number of shorter jobs.

Running a large simulation on the Grid

Next, we selected a job size that would ensure all jobs would run within the minimum queue lengths. Thus, instead of simulating 100 independent cycles for each set of initial conditions, we ran 10 jobs of 10 cycles, each requiring between 8 minutes and 8 hours.

We then changed job management to launch all the jobs at once and use a daemon that would periodically check job status, retrieve results, if complete, or resubmit if aborted, looping for a reasonable time to ensure all jobs had a chance to terminate. With the new approach,

we achieved success rates of 90% and analysed the rest to determine the reasons for failure.

The most concerning kinds of failure were *unspecified job failures*. As there is very limited information on these failures, and they are relatively infrequent, there is little else to be done besides re-starting them. A special kind of problem that appears about one in every 9,000 jobs is that *job submission hangs indefinitely*. A more worrisome anomaly is *immortal jobs*. These are jobs that remain in 'Running' status indefinitely, even after Grid-execution permissions have expired, probably because the job termination notification has been lost. Finally, we were made aware of a side-effect of our approach on other users: while we had reduced the load on our front-end (the User Interface or UI node), we were using and overloading our default Grid Resource Broker (RB), which takes care of matching jobs to available resources. As the RB is shared among several sites, our load was affecting many other users. Other failures identified involved successful jobs whose output was lost, unrecoverable or empty.

To solve submission problems, we extended our submission tool to use a time-out to detect stalled submissions, and to maintain a dynamic list of available RBs to load-balance submissions over them and avoid overloads. As for job failures, we added to the monitor script the ability to detect aborted or failed jobs and to resubmit them automatically. This simple device is useful for most problems except immortal jobs, which can only be detected if it is possible to impose an upper bound on execution times that may be used as a time-out or, if not, by submitting jobs more than once to collect the results of the first to finish, and kill the others.

Efficiency measures

Using the timings collected, we could measure for each job the time spent on submission ($s_1 - s_0$), time required by the Grid to allocate resources and start the job ($e_0 - s_1$), time taken by the job ($e_1 - e_0$), delay incurred to detect job termination ($r_0 - e_1$), and time needed to retrieve results ($r_1 - r_0$). In addition, by collating the individual statistics, it was easy to measure total times incurred at each step: e.g., for submission, it would be $\max\{s_1\} - \min\{s_0\}$, accumulated CPU time ($\sum(e_1 - e_0)$), total execution wall-clock time ($\max\{r_1\} - \min\{s_0\}$), etc.

The mean **execution time** for our jobs varied slightly across experiments, about 8-10K seconds,

yielding, in principle, a good balance between the serial and parallel parts. However, time variation ranged between ~500 and 115,000 seconds.

Our initial estimation of the benefit expected from the Grid was based on our perception that **job submission** was a quick process, which we further bound with a time-out. Indeed, our measures reveal that, for our problem (homogeneous jobs of ~800KB in size), submission times are in the range of 12-266 seconds, with a mean of 32 seconds. Thus, the contribution of the submission step is very low in relation to the average running time (0.3-0.4%). Something similar happens with the final **output retrieval** step, which ranges between 5 and 150 seconds.

There are other sources of overhead though: once a job is copied to the Grid, there is a delay owing to **internal Grid housekeeping**. Similarly, once a job is finished, there is a delay until the overall Grid self-monitoring structure gets notified and the status is updated.

From our measures, we conclude that this contribution is significant and poses a strong tax on the efficiency gains that can be achieved: the time taken for a job to start execution ranged between 30 seconds and 60K seconds, with an average of ~4-6K.

In order to put these measures in perspective, we need to know the **number of CPUs actually used**: we noted the host name of the WNs and counted the number of different machines accessed for each simulation experiment. Usual numbers were uniformly around 2,400 different machines for a simulation running 10,000 jobs.

Finally, by comparing the actual execution time of the job with the total wall-clock time taken, we can quantify efficiency gains: on average, jobs took ~9 times longer to run on the Grid, with the best case taking only 1.006 and the worst case 150 times more than local execution.

The massively parallel nature of the Grid, however, may compensate for these efficiency losses by allowing many jobs to run simultaneously. We added the total CPU time used for a 10,000 job experiment and divided it by the total time taken. This total time includes job resubmission and hence accounts for more than 10,000 actual jobs. For our problem, this consistently resulted in a speed-up of ~190-fold relative to a single computer.

To quantify these benefits, let us denote N_n the number of nodes used, N_j the number of jobs to be run, t_j the time per job, t_s the time to submit a job, t_b the time used in Grid house-keeping tasks, t_e the execution time, and t_r the time required for result retrieval.

- (1) The average time needed to run a job would be $t_j = t_s + t_b + t_e + t_r$.
- (2) The time needed for sequential execution of our jobs on a single node would be $t_1 = t_e \times N_j$, whereas the time needed for sequential execution on the Grid (e.g., using only one node) would be $t_g = t_j \times N_j$, which, as $t_j > t_e$, means that Grid execution time is obviously longer for sequential jobs.
- (3) The time required for parallel execution on the Grid is more difficult to evaluate, and depends on the number of nodes that can be used in parallel. Ideally, the Grid overhead times (t_s , t_b and t_r) should be close to zero, making the total time for parallel execution $\rightarrow t_1 / N_n$. Ideally, one would expect nodes to be reconsidered as soon as they finish a job, hence $N_n \propto (t_e / t_s) + 1$. However, as the Grid is geographically spread, one may expect a significant delay between the time a node finishes execution and the time an RB notices it is free. This has an impact on resource allocation, which now takes longer, making $N_n \propto ((t_b + t_e) / t_s) + 1$. This means that we may expect to use up fewer nodes for short-running jobs than for long-running jobs. We may also derive estimations for the maximum number of nodes that can be reached by using the maximum values of t_b and t_e and the minimum value of t_s .

We have already seen that both $\overline{t_s}$, and $\overline{t_r}$ are relatively small (~30 seconds each), and thus, as $t_e \gg t_s \wedge t_r$, their impact tends to zero (0.3 – 0.4% in our case). The scheduling overhead, however, is non-negligible. This delay becomes significant for small job numbers and for short jobs, hence reducing Grid speed-up². On the other hand, as execution time decreases, the impact of the time required for sequential job submission increases. This can be ameliorated

² We have been able to verify these results on other kinds of problem with different numbers of jobs and execution times (Carrera, G., Solano, A., Valverde, J. R. and Carazo, J.M., unpublished).

by partially parallelising job submission, but will still hit a sequential limit in data transfer from the submission node to the RB, and usually results in downgraded performance with respect to an ideal parallel execution.

Discussion

We needed to reproduce the behaviour of a population system whose experimental analysis would have been too cumbersome to simulate fully, being a stochastic process (mutations), which requires Monte Carlo-like methods. The dynamic behaviour of the system results in dramatic population size changes, depending on the initial parameters (as a higher impact on survival fitness means slower growth and smaller populations), which in turn results in a wide variation in running times (various orders of magnitude).

The Grid gives any researcher immediate access to huge computing power through a large number of geographically spread machines. For large parallel problems with reduced communication needs such as this, the Grid is an easy and powerful solution.

Optimising computation

Communications in the Grid have a larger latency and are slower than on a cluster; hence, it is desirable to keep them at a minimum in relation to parallel computation, according to Amdahl's law. The best trade-off can be achieved when computation may proceed for long times with a large number of jobs, but most sites impose run-time limits (usually 72h).

If the number of jobs to perform is not too high, users may aim for the smaller number of sites that accept longer jobs on their queues. On the other hand, if users prefer to get results more swiftly by splitting the work among many shorter jobs, the number of available machines increases considerably.

When execution times are fairly homogeneous, users may fine-tune jobs to fit on the allowed time-slot and optimise the communications/computation ratio; in our case, large run-time variability forced us to plan for the worst-case scenario (ensure longest jobs would fit), resulting in relevant efficiency penalties for the shortest jobs.

Job management

For running a single job, the EGEE Grid offers convenient commands for the user. However, when the number of jobs grows to the order of thousands, new problems arise that demand more sophisticated job-handling mechanisms: the incidence of aborted or failed jobs, for various reasons, may reach 10-15% of jobs, requiring the inclusion of additional job-management procedures. The most immediate approach, and the one we have used here, is to detect and re-start failed jobs up to a maximum number of times, but other approaches are possible: e.g., launching various instances of the same job, taking the results of the first to finish and discarding all others, or waiting for various jobs to finish and comparing their output for additional resilience.

As the number of jobs increases into the tens of thousands, new issues need to be considered. First, we reduced overload over the RB by performing some load balancing over all available hosts. As RBs themselves may also fail, a dynamic detection and recovery mechanism for failing RBs was added too. Second, very rare events need to be considered and dealt with, either manually (if their incidence is low enough and circumstances allow) or automatically. The most relevant of these is probably jobs hanging on submission, as this may stop the whole experiment; stalled submission can be conveniently dealt with by implementing a simple time-out mechanism.

A different problem is posed by immortal jobs, which remain eternally in 'running' state. This may be easy to spot if upper-bound estimation of job run-time is possible, so that jobs exceeding it can be considered lost and re-started; but when there is high variability in run-times (as was our case), or there is no easy way to predict an upper bound, detection of these jobs becomes increasingly difficult, as the long run-time might be inherently correct. In such cases, possible solutions are:

- run single instances and after sensible time (we used ~80 hours) detect, kill and re-start unfinished jobs;
- replicate all jobs and take results from the first to finish, killing all other copies.

Efficiency considerations

We have taken timing measures at the various steps *avoiding use of our local cluster and making sure jobs were freely allocated to any WNs by the Grid*, so that measures include real-world effects. Timing checkpoints were taken using UTC to enforce a common time frame.

Regarding Grid efficiency, we can see that the submission process is efficient. The same can be said of result retrieval. Consequently, their impact is almost negligible. This is demonstrated by our finding a minimum efficiency loss of 0.006 for a Grid job not executed on our local cluster. Once the job is submitted, jobs suffer a house-keeping delay until execution. In our experience, job scheduling took a significant amount of time (on average, 4-6K seconds) with large variability. Given our experiment design, we did not take accurate measures of Grid house-keeping after jobs finished: it is possible that there were large delays, which we didn't detect because our data were actually available when we performed the test. Nevertheless, our results suggest that this final step may be fairly quick, taking perhaps a few minutes, but this needs confirmation.

With these data at hand, we can already draw several conclusions, which can be used as advice for Grid usage. First, resource management on the Grid is undoubtedly the area where biggest efficiency gains can still be achieved. If efficiency is a concern, it may be worth considering using alternate scheduling mechanisms, such as those provided by GridWay (Huedo *et al.*, 2004), currently part of the Globus Toolkit (Foster and Kesselman, 1997) and planned for inclusion on [glite](http://glite.cern.ch)³.

For single jobs, efficiency may reduce to as little as 1.006 or as much as 150 times; however, on average, it will be reduced by about one order of magnitude. Thus, if the single job to be run is a Message Passing Interface (MPI) parallel job to be launched against a big (more than 10-node) cluster, it may compensate for the Grid inefficiency. If the job takes too long and the system cannot be tied for that amount of time (e.g., a shared desktop), or if the local system is already overloaded (e.g., a time-sharing system with too many CPU-bound processes), then the Grid provides a convenient way to run jobs that otherwise would be impossible, difficult or very slow to complete locally.

For large numbers of jobs, the Grid provides a way to speed up problems and deliver quicker responses, which may prove successful for most researchers. For instance, we were far from the maximum theoretical linear speed-up (10,000 times for 10,000 independent processes), and even from the practical speed-up (2,400 times for the 2,400 different CPUs we could harvest), but we still could accelerate our problem 190 times, which allowed us to run in 1½ days (1 day 14h 01m 42s) a project that otherwise would have taken almost one year (313 days 04h 39m 33s), or in 4½ days (4 days 19h 38m 37s) a project requiring 2½ years (930 days 02h 25m 20s) of CPU time.

It is worth noting that our low efficiency was partly the result of our unequal run-times, which prevented reaching a better parallel/serial ratio. Higher speed-ups should be possible for better-behaved problems, or with more refined job-management strategies.

Conclusion

We have been able to run large-scale population dynamics simulations on the Grid with relatively little effort: no changes were needed to the simulation software, work was split into suitably-sized chunks for execution, and job management was handled by relatively simple shell scripts. In the process, we had to deal with and solve a number of problems, developing generic tools that are available under the [GNU public license](http://www.gnu.org/licenses/public.html)⁴ from the author.

Each experiment involved large numbers of jobs (usually 10,000), allowing us to collect statistical data to monitor Grid performance and efficiency gains. We have identified Grid house-keeping as a major contributor to reduced efficiency, although we could still achieve significant speed-ups (~190x) using thousands (>2,400) of CPUs, allowing us to solve in days a problem that would otherwise have taken years to complete. Our results are in line with observations on other applications by our group and others (Jacq *et al.*, 2007), and lay the basic foundation for understanding the main issues affecting Grid development for large embarrassingly parallel applications.

3 <http://glite.cern.ch>

4 http://ahriman.cnb.csic.es/sbg/tiki-list_file_gallery.php?galleryid=1

Acknowledgements

The author wishes to acknowledge the invaluable scientific cooperation of A. Couce and J. Blázquez of CNB/CSIC.

Funding: I wish to thank the European Commission for its support to projects EGEE (INFSO-RI-031688) and EMBRACE (LHSG-CT-2004-512092), which made this work possible.

References

1. Adami C, Ofria C, Collier TC (2000) Evolution of biological complexity. *Proc Natl Acad Sci USA* **97**, 4463-4468.
2. Foster I, Kesselman C (1997) Globus: A metacomputing infrastructure toolkit *Int J Supercomput Appl* **11**(2), 115-128.
3. Huedo E, Montero RS, Llorente IM (2004) A framework for adaptive execution in Grids. *Softw Pract Exper* **34**(7), 631-651.
4. Jacq N, Salzemann J, Jacq F, Legré Y, Medernach E *et al.* (2007) Grid-enabled Virtual Screening Against Malaria. *J Grid Computing* **6**(1), 29-43.
5. Lenski RE, Ofria C, Collier TC, Adami C (1999) Genome complexity, robustness and genetic interactions in digital organisms. *Nature* **400**, 661-664.
6. Sniegowski PD, Gerrish PJ, Lenski RE (1997) Evolution of high mutation rates in experimental populations of *E. coli*. *Nature* **387**, 703-705.
7. Taddei F, Radman M, Maynard-Smith J, Toupance B, Gouyon PH, Godelle B (1997) Role of mutator alleles in adaptive evolution. *Nature* **387**, 700-702.
8. Wilke CO, Wang JL, Ofria C, Lenski RE, Adami C (2001) Evolution of digital organisms at high mutation rates leads to survival of the flattest. *Nature* **412**, 331-333.

a balanced mind

Vivienne Baillie Gerritsen

When I leave for work every morning, I know exactly where to get my train. This may sound quite absurd but just imagine, for one moment, that you had no memory. You would always be losing your keys. You would never remember where you had left your shoes. And you'd probably fall down the front doorstep daily because you had forgotten there was one. Thanks to our faculty for memorising things, life is far easier for us. We learn how to talk. We learn to avoid awkward situations. We even remember who our children are. On the molecular front, there is a lot going on. It all has to do with neurons and their ability to pass on messages and connect to one another. Unsurprisingly, many proteins are involved in the processes of learning and memory, and much research has been done on them in the past years. There is one protein, however, known as RGS14, which is a bit of a conundrum. Indeed, RGS14 seems to have the intriguing role of suppressing memory...



by PET

Courtesy of the artist

Deliberately suppressing the ability to remember something may sound unreasonable. Yet the art of forgetting is also important. We have to forget all the words we hear throughout the day. We have to forget all the prices we see on a restaurant's menu. We have to forget all the faces we brush past as we rush across town. Our brain needs to filter the hundreds of thousands of messages we bump into every day. If it doesn't, we would all be on the verge of madness. Memory is thus a question of balance between remembering some things and forgetting many others.

The notion is not new. There is a psychiatric disorder known as the Savant Syndrome* caused by the malfunction of a phosphatase, PP1, which – in natural circumstances – hinders

the synthesis of proteins involved in memory. Those inflicted with the disorder are submerged with useless information they are unable to forget. Hence, the importance of a basic memory filter. So why all the fuss about RGS14? Because RGS14 not only belongs to a part of the brain which, until now, had shown no involvement whatsoever in the memory process but also because when it is shut off, memory seems to be enhanced without any side effects. Which sounds like magic...

Current wisdom suggests that, in the brain, the seat of memory and learning is situated in the hippocampus. Until recently, one small region known as CA2 had been neglected by researchers because – unlike the rest of the hippocampus – it didn't seem to have any say in memory. But it turns out that it does, in a certain sense. Indeed, CA2 is full of RGS14. So, yes, in natural circumstances, RGS14 suppresses the faculty of memorising. But when the protein was silenced in mice, scientists discovered that the rodents were not only intrigued by new objects – thus meaning that they had recognised pre-existing ones which were consequently of less interest – but they were also far brighter than their wild-type companions at making their way through a maze.

So what is happening on the molecular level? The answer is synaptic plasticity. Memory is believed to be a case of synaptic transmission between neurons, and the strengthening of such

connections. This has been termed synaptic plasticity and forms the basis of acquiring and consolidating certain forms of learning and memory. These processes are known to occur in the hippocampus, save in the CA2 region. Which is one of the reasons this region had been ignored until now. So, if synaptic plasticity is at the heart of memory, how does RGS14 act upon it?

RGS14 belongs to the very large family of G protein signalling regulators (RGS) and directly suppresses the activity of a certain number of proteins whose downstream effects would otherwise be crucial in the processes of learning and memory. More specifically, RGS14 binds to G proteins as well as to components of the mitogen-activated protein (MAP) kinase signalling pathway – both of which are required to strengthen synaptic transmission. When the effects of RGS14 are wiped out in mice for example, G protein and MAP kinase signalling pathways are free to be activated, synaptic plasticity is restored and the rodents' capacity to

remember objects is enhanced. Thus making them somewhat smarter than they otherwise were expected to be.

What is more, putting a rein on RGS14 doesn't seem to have any side effects on the mice's psyche. For as much as one can really measure such a subtle state of things. But, once again, a mouse is not human, and there is a great chance that RGS14 is part of our brain – or a rodent's – for a reason other than memory. To be sure, the rest of the hippocampus does that... Perhaps RGS14's faculty of suppressing memory is just a side effect of something far more important it can do that we are unaware of. After all, the loss of neurons in the CA2 region is known to be involved in psychiatric disorders such as schizophrenia for instance. This said, RGS14 is restricted to CA2, itself a discrete region of the hippocampus, which makes the protein an ideal candidate for the future design of therapeutic agents that could ease psychiatric disorders. Or simply help to diminish the increasing ease with which we forget things over the years.

**N.B. Also read Protein Spotlight issue 32, "The things we forget"*

Cross-references to UniProt

Regulator of G protein signaling 14 (RGS14), *Mus musculus* (Mouse) : P97492

Regulator of G protein signaling 14 (RGS14), *Homo sapiens* (Human) : O43566

References

1. Vellano C.P., Emerson Lee S., Dudek S.M., Hepler J.R.
RGS14 at the interface of hippocampal signaling and synaptic plasticity
Trends in Pharmacological Sciences [Epub ahead of print] (2011)
PMID: 21906825
2. Emerson Lee S., Simons S.B., Heldt S.A., Zhao M., Schroeder J.P., Vellano C.P., Cowan D.P., Ramineni S., Yates C.K., Feng Y., Smith Y., Sweatt J.D., Weinshenker D., Ressler K.J., Dudek S.M., Hepler J.R.
RGS14 is a natural suppressor of both synaptic plasticity in CA2 neurons and hippocampal-based learning and memory
PNAS 107:16994-16998(2010)
PMID: 20837545
3. Shu F.-j., Ramineni S., Hepler J.R.
RGS14 is a multifunctional scaffold that integrates G protein and Ras/Raf MAPkinase signalling pathways
Cellular signaling 22:366-376(2010)
PMID: 19878719

National Nodes

Argentina

IBBM, Facultad de Cs.
Exactas, Universidad
Nacional de La Plata

Brazil

Lab. Nacional de
Computação Científica,
Lab. de Bioinformática,
Petrópolis, Rio de Janeiro

Chile

Centre for Biochemical
Engineering and
Biotechnology (CIByB).
University of Chile, Santiago

China

Centre of Bioinformatics,
Peking University, Beijing

Colombia

Instituto de Biotecnología,
Universidad Nacional de
Colombia, Edificio Manuel
Ancizar, Bogotá

Costa Rica

University of Costa
Rica (UCR), School of
Medicine, Department
of Pharmacology and
ClinicToxicology, San Jose

Finland

CSC, Espoo

France

ReNaBi, French
bioinformatics platforms
network

Greece

Biomedical Research
Foundation of the Academy
of Athens, Athens

Hungary

Agricultural Biotechnology
Center, Godollo

Italy

CNR - Institute for Biomedical
Technologies, Bioinformatics
and Genomic Group, Bari

Mexico

Nodo Nacional de
Bioinformática, EMBnet
México, Centro de Ciencias
Genómicas, UNAM,
Cuernavaca, Morelos

Norway

The Norwegian EMBnet
Node, The Biotechnology
Centre of Oslo

Pakistan

COMSATS Institute of
Information Technology,
Chak Shahzaad, Islamabad

Poland

Institute of Biochemistry and
Biophysics, Polish Academy
of Sciences, Warszawa

Portugal

Instituto Gulbenkian de
Ciencia, Centro Portugues
de Bioinformatica, Oeiras

Russia

Biocomputing Group,
Belozersky Institute, Moscow

Slovakia

Institute of Molecular Biology,
Slovak Academy of Science,
Bratislava

South Africa

SANBI, University of the
Western Cape, Bellville

Spain

EMBnet/CNB, Centro
Nacional de Biotecnología,
Madrid

Sri Lanka

Institute of Biochemistry,
Molecular Biology and
Biotechnology, University of
Colombo, Colombo

Sweden

Uppsala Biomedical Centre,
Computing Department,
Uppsala

Switzerland

Swiss Institute of
Bioinformatics, Lausanne

Specialist- and Assoc. Nodes

CASPUR

Rome, Italy

EBI

EBI Embl Outstation, Hinxton,
Cambridge, UK

Nile University

Giza, Egypt

ETI

Amsterdam, The Netherlands

IHCP

Institute of Health and
Consumer Protection, Ispra.
Italy

ILRI/BECA

International Livestock
Research Institute, Nairobi,
Kenya

MIPS

Muenchen, Germany

UMBER

Faculty of Life Sciences, The
University of Manchester, UK

CPGR

Centre for Proteomic and
Genomic Research, Cape
Town, South Africa

The New South Wales Systems Biology Initiative

Sydney, Australia

for more information visit our Web site

www.EMBnet.org

EMBnet.journal

ISSN 1023-4144

Dear reader,

If you have any comments or suggestions regarding this journal we would be very glad to hear from you. If you have a tip you feel we can publish then please let us know. Before submitting your contribution read the "Instructions for authors" at <http://journal.EMBnet.org/index.php/EMBnetnews/about> and send your manuscript and supplementary files using our on-line submission system at <http://journal.EMBnet.org/index.php/EMBnetnews/about/submissions#onlineSubmissions>.

Past issues are available as PDF files from the Web site:

<http://journal.EMBnet.org/index.php/EMBnetnews/issue/archive>

Publisher:

EMBnet Stichting
c/o Erik Bongcam-Rudloff
Uppsala Biomedical Centre
The Linnaeus Centre for Bioinformatics, SLU/UU
Box 570 S-751 23 Uppsala, Sweden
Email: erik.bongcam@bmc.uu.se
Tel: +46-18-4716696