Computational Chemistry, Research, and the Computer: Preparing a Computer for Quantum Mechanical - Molecular Mechanical Simulations

By Jonathan B. Miller, Advisor: Dr. Pedro Muíño
Department of Chemistry, Saint Francis University, Loretto, PA 15940

April 4, 2008

Work was performed on two Silicon Graphics computers in order to use the first computer as a reference system and aid in developing the second computer as a molecular modeling system. Open source versions of FORTRAN and C++ compilers were installed and a full operating system update to IRIX 6.5.2 was performed on the second computer. The CHARMM modeling software was unsuccessfully installed due to software incompatibility. LaTeX-based typography software was installed on a third computer but was deemed too involved for the scope of the project. Molecular research was performed on an insulin protein to illustrate the concepts of quantum mechanical – molecular mechanical simulation like that to be used in researching α-conotoxin GI.

I. Introduction, Background, & Relevant Problems

Computational chemistry has been increasing in popularity in recent decades as computer processing power is increasing in both amount and availability. This discipline of chemistry focuses on using computation to help solve chemical problems. Based on theoretical chemical theories, computer programs are written that can simulate and predict interactions of atoms or molecules to supplant or (in the case of this research project) aid traditional laboratory chemistry.

The goal of this research project was to develop a computer system that can be utilized for molecular modeling simulations in order to support Dr. Pedro Muíño on his collaborative research with Dr. Balazs Hargittai.[1] Both researchers are chemistry professors at Saint Francis University. There are two Silicon Graphics computers in the chemistry department upon which efforts were focused and an additional properly-configured remote computer at Montana State University that serves as a 'fail-safe' if the SFU computers are not functioning as desired. The chemical simulations intended to be performed were the molecular dynamics on α-conotoxin GI with the basic amino acid arginine attached at position 9 (Figure 1) and the variations of arrangements that are due to mispaired unfolding disulfide bridge regioisomers. The different regioisomers have varying combinations of disulfide bridges because there are three ways to form a pair of disulfide bridges.

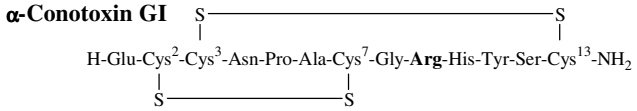

Figure 1. Simplified structure of α-conotoxin GI, the protein under study.

The computer development process necessitated knowledge of and experience with a variety of topics, including quantum mechanical and molecular mechanical (QM-MM) modeling, operating system debugging and maintenance, programming, compiling, server networking, and technical writing and presentation.

The chemistry department provided two computers for research use, both of which have unique individual personalities (as much as a machine can have, at least). The oldest of the pair, referred to as "*Breogán*", is an SGI O_2 running IRIX 6.4, a proprietary operating system. QM-MM simulations have been performed on it in the past. The computer was developed and formerly maintained by a technical staff at Kansas State University prior to its use at SFU. Along with aging hardware and software, *Breogán* has very little free storage space available, and there is no clear way to install additional SCSI hard drives (internal or external) to gain more storage space. *Breogán* has a CD-ROM drive, making it the only SFU computer within the department that can 'read' the special SGI IRIX compact disks that are needed to maintain the operating system and install additional software. A primary issue with *Breogán* was that it stubbornly refused to boot to the operating system.

The second computer in the chemistry department, named "*Galicia*", is an SGI Octane running IRIX 6.5. *Galicia* was the computer that was intended to be developed as the QM-MM simulation machine, but this function necessitated the installation of various types of software, of which *Galicia* had none (it had an operating system and little else). There is no CD-ROM drive, which impedes file transfers to the system.

As mentioned previously, both of these computers are products of Silicon Graphics, Inc. Both computers run IRIX, which is SGI's proprietary operating system that is based on the very stable UNIX kernel and is similar to Red Hat Linux, Fedora Core, and Ubuntu. Additionally, both machines have high-quality, optimized hardware, such as the MIPS central processing unit, that is specifically designed for 'number crunching' operations. These qualities make the computers very useful for mathematically-intensive molecular simulations, more so than many of today's powerful but un-optimized personal computers.

When possible, free, open-source software was to be used for the programming compilers and for other uses. This type of software is useful because of the lack of a price tag and the relative quality of product. However, free software generally comes with only adequate support resources, such as installation tutorials or user forums and online bulletin boards, but certainly no toll free help hotlines, for example. Sometimes pay-software is preferred or the only option available. FORTRAN is the programming language required for the molecular simulation software.

The molecular modeling software that was to be installed and used was CHARMM (Chemistry at HARvard Molecular Mechanics). The software licensing had previously been acquired. CHARMM is both the name of a commonly-used set of molecular dynamic force fields and of a software package that employs the fields.

There was documentation available to work with CHARMM and to assist in conducting the research project. Online documentation, reference manuals, and internet-based public support forums exist for some of the other software that was expected to be utilized.

II. Experimental

Before trying to work directly with *Breogán* and *Galicia* it was necessary to perform some basic research into the directory structure, commands, and overall operating environment of UNIX systems. The text *Exploring the UNIX System* provided many examples of how the command line interface (CLI) can be utilized to perform all functions on the system without using a graphical

user interface (GUI) which is what most common users work with but is sometimes limited in capability.[2] The CLI text editor, *vi*, was needed to work with text and programming files, so familiarity with this software was aided by the text *Mastering the vi editor*.[3]

The next need was to perform a successful boot of *Breogán* in order to use it as the 'reference computer' when setting up *Galicia*. When the power button was pressed on *Breogán*, a red light and an internal cooling fan would turn on but nothing would display on a monitor. Upon investigation of the internal and external components of the computer and after trial and error, it was discovered how the display adapter can be led to work about 80% of the time: To successfully boot *Breogán* and ensure that information is displayed on a monitor, press the power button and wait several seconds, then pull the cord from the power supply (cutting power to the computer), then plug the cord back in, and finally press the power button again. This procedure should invoke the display adapter to function properly so *Breogán* can be used as a molecular modeling reference system.

With *Breogán* resurrected, both it and *Galicia* were moved to a research lab. Since much of molecular modeling is based on programming with text documents, and since the researchers wanted the capability to 'telecommute' to the research lab, *Breogán* and *Galicia* needed to be connected to the school's local network. Static internet protocol (IP) addresses were reserved by SFU's Information Technology Services for use by the pair of computers. A computer's IP address serves as the computer's unique identifier on a digital network. *Breogán* was assigned the address of 10.0.2.12 and *Galicia* was assigned 10.0.2.11. Since the computers have network access, they can be accessed by any other computer on the network by using Telnet or FTP protocols, for CLI operations or file transfers, respectively. As a secure alternative to Telnet, Secure Shell could be used.

Having both systems performing basic operations successfully, the next phase of the project was to attempt to outfit *Galicia* as a fully-functioning molecular simulation machine. Based on knowledge of how *Breogán* performed its simulations, all that was known about how to prepare *Galicia* was that there needed to be a functioning compiler, by default MIPSPRO, in order to have the molecular modeling software, CHARMM, work. Since *Galicia* was a 'fresh out of the box' computer, it was clear that CHARMM would need to be installed since it is a third-party software package. It was expected that the MIPSPRO compiler would be installed on the system since internet research showed that every SGI Octane system comes with the compiler as part of the initial purchase price. Anonymous sources indicated that the result of the 'cc' command would indicate the presence of a compiler. When performed on *Galicia*, an error was reported, suggesting that a compiler was not installed or not functioning – neither of which seemed plausible since the IRIX OS was a fresh install and should come with a default compiler. A support forum recommended downloading the Multiple Precision Floating-Point Reliable Library[4] to try to fix the broken 'cc' command. A suitably-packaged IRIX-specific version[5] was downloaded and extracted. However, the contents had to be compiled on the system before they could be used – which was impossible since it was the compiler that the MPFPRL files were supposed to fix. Further research led to the following failed attempts: (1) A bootstrapper program[6] was acquired that could have potentially addressed the compiler issue but the program had little documentation on how to use it; (2) IRIX equivalents of RPM (Red hat Package Manager) packages or YUM (Yellowdog Updater, Modified)[7] software were sought but could not be found; (3) Porting software from NetBSD, a UNIX distribution similar to IRIX, was time-intensive and was not well-documented or guaranteed to work.

Since the MIPSPRO compiler was non-existent, FORTRAN and C++ compiler installations were attempted. However, the specific compiler versions were meant to be used on IRIX 6.4, whereas *Galicia* had IRIX 6.5, rendering the compilers incompatible. Since FORTRAN and C++ compilers are more commonly used than a MIPSPRO compiler, open-source software was considered. The GNU Compiler Collection (GCC) is a suite of free, well-performing, and portable compilers that would allow FORTRAN and C++ programs to be compiled as if professionally-made, expensive compilers were utilized. Typically, the GCC package would be downloaded and compiled prior to installation, which would have rendered it unusable on *Galicia* since this computer lacked any compilers. However, research located a website called TheWrittenWord.com that provides free, pre-compiled binaries of GCC 4.0.2 for IRIX 6.5. The GCC binary was downloaded, extracted, and then installed. Unfortunately, when trying to compile test programs using the commands `./gcc test.c -o test` and `./gfortran test.f -o test1` to test the GCC's C++ and FORTRAN compilers, respectively, syntactical errors were reported, indicating an improperly-installed GCC suite.

To solve this problem, a large IRIX 6.5 update package called an Overlay was installed. The Overlay consists of four primary compact disks and one IDL (IRIX Development Libraries) compact disk. Four issues hindered a quick installation of the Overlay: (1) Since *Galicia* has no CD-ROM drive there is no simple way of installing the disks' content; (2) The Overlay disks cannot be read by a non-IRIX OS, reducing the number of computers able to read the disk to just *Breogán*; (3) An FTP server could be used to transfer the disks' content from *Breogán* over the network to *Galicia*'s hard drive in the form of a disk image except *Breogán* was not receptive; (4) *Galicia* had about one gigabyte of free hard drive space so there was room for just two disk images. An improvised solution was to place the Overlay1of4 disk into *Breogán*, copy the disk's content via FTP onto a 'middleman' Windows XP laptop, then transfer the disk image by FTP to *Galicia*, then execute the image on *Galicia* in order to begin the Overlay installation. The IRIX hardcopy instruction manual recommended having all additional Overlay disks ready during the installation process. However, there was not enough hard drive space left to move a second disk image onto *Galicia* since a typical disk image is five hundred megabytes. An external SCSI (Small Computer System Interface) hard drive was not available.

The Overlay installation began by transferring the Overlay1of4 disk image to *Galicia* and then opening the image with Software Manager, IRIX's built-in automated software installer. After Overlay1of4 was installed, Software Manager asked to open the next disk image, so Overlay2of4 was transferred to *Galicia*. After the second image was complete, it was deleted from *Galicia*'s hard drive and Overlay3of4 and then Overlay4of4 were moved over. Ninety-seven file version conflicts were detected. One of Software Manager's recommended solutions to the conflicts was to install the IRIX Development Libraries (IDL) disk so Overlay4of4 was deleted and IDL was moved to *Galicia*. Installation continued but terminated from lack of available hard drive space. There were only one hundred megabytes of free space into which the ten times larger, gigabyte-size Overlay was to be installed. The lack of hard drive space resulted from the presence of two disk images being kept on the hard drive to facilitate the installation.

A potential remedy for the hard drive requirements was to use the networking capabilities of *Breogán* and *Galicia* to configure *Galicia* to use *Breogán*'s CD-ROM drive as if the drive was installed in *Galicia*. A shortcut link was placed on the desktop of *Galicia* that pointed to *Breogán*'s IP address. However, when the shortcut link was tested using both HTTP and FTP protocols to

establish the connection between the two computers, the link failed due to routing issues. It is suspected that the lack of a Network File System server on the SFU network was the reason for the two computers to be incapable of communicating.

Returning to the procedure of 'rotating disks' to install the Overlay, one disk image at a time was placed on *Galicia*'s hard drive, the installation was initiated, and when another disk was needed for installation to continue the first was deleted, and so on. After an eight hour procedure that involved moving two dozen disk images to and from the hard drive, the installation was ninety percent complete but then halted. The Software Manager reported that several files were corrupt in the archives. The "ignore" command was sent and the installation completed without further issue.

With the OS updated from IRIX 6.5.0 to IRIX 6.5.2, compiling commands were tested again and found to be working. The GNU Compiler Collection was working properly and the next phase of the research project could begin: installing CHARMM, the molecular modeling software.

The version of CHARMM to be installed was an academic edition called "c32b2". Installation instructions were provided with the software, which was meant to be easy to install. To initiate the installation, two scripts could be executed: `install.com sgi` or `install.com gnu`. The former script was executed but errors were reported; all of the installation files were deleted and begun anew, executing the latter script, resulting in the same errors.

The installation failed because of "`g77 - Command not found`", followed by "`The CHARMM executable /usr/people/jonathan/c32b2/exec/gnu/charm is Not produced.`" It turns out that g77 was an antiquated GNU FORTRAN compiler that is no longer maintained and was replaced with gfortran, which was installed on *Galicia* via the GCC. CHARMM version c33 included support for gfortran.[8] Also, a user on the CHARMM support forums said that "I'm not sure that g77 is compatible with gcc 4.x".[9] A post was made on the forums that described the situation in hopes of leveraging the community's familiarity with the software to devise a solution.[10] User 'rmv' replied that "the error is coming from install.com, where it attempts to build the pre-processor in the tool subdir using g77 for the 'gnu' machine type." Rmv said that the error can be bypassed by "manually compiling prefx.f with gfortran and naming the executable prefx_gnu". This was performed by typing "`gfortran prefx.f -o prefx_gnu`" in the tool/ directory and then "`./install.com gnu`".

The g77 error did not occur but the CHARMM executable was still not produced. A syntax error originated in the Makefile code of build/gnu/ at line 15: "`ADDLIB := $(ADDLIB) $(GLIB)`". User 'rmv' recommended replacing "`:=`" with "`=`" to fix the error but this led to more syntax errors.

These particular errors are indicators that the SGI "`make`" command was being called instead of the GNU "make" command, which is acquired by installing a program named "gmake". The software was acquired and installed but the installation errors were the same, despite tweaking the Makefile code and setting up alias commands. It was discovered that using the "setenv MAKE_COMMAND gmake" in the terminal would cause the CHARMM errors to say "`gmake - Command not found`" instead of "`make - Command not found`". Progress was being made since the syntactical errors were occurring for gmake, not make – a step in the right direction. Further work on installing CHARMM was postponed in favor of initiating molecular modeling research on another computer system, *Marcus*, at Montana State University.

RasMol[11], a molecule viewer program, was installed to view a test file of an insulin protein, downloaded from the Research Collaboratory for Structural Bioinformatics Protein Data Bank (RCSB PDB)[12]. A tutorial file on *Marcus* was used to aid in the molecular research of the insulin test protein. When *Marcus* fatally broke, a similarly-equipped computer named *Zerner* was used for research.

The modeling procedure began by acquiring a Protein Data Bank (PDB) file from RCSB PDB. The PDB file contained atom and residue names (such as carbon or nitrogen and alanine or histeine, respectively), coordinates, and segment arrangements that described how the atoms and residues were specifically attached together to form an insulin protein structure. The 'raw' PDB file had to be converted by script to a pair of PSF and CRD files while also solvating the molecule with water at a twenty-five angstrom radius. The PSF file held information about the positions of bonds, which describe how atoms were connected to each other. The CRD file noted the molecular topology - the three-dimensional coordinate position of every atom of the molecule. Another script was then used to invoke CHARMM to open the PSF and CRD files and perform the molecular dynamics simulation using a CHARMM force field.

After modeling was performed, a paper had to be composed to document the research project. The LaTeX typesetting software was intended to be used to produce the paper because LaTeX is commonly used for publishing scientific documents. A Microsoft Windows implementation of LaTeX called proTeXt was downloaded and installed.[13] However, there were complications in configuring the software to run properly. An online-accessible LaTeX webservice, MonkeyTeX, was tested but Microsoft Word was deemed as most practical because the learning curve for MonkeyTeX was too great to justify its use for the paper.[14]

III. Results

As with many computer development projects, unforeseen issues cropped up during research that threatened to halt or hinder portions of the project. Fortunately, many of the project's original goals, ranging from getting one computer to simply boot to installing functioning compilers to performing molecular modeling research, were accomplished.

The original molecular modeling computer, *Breogán*, was restored to a condition of full-functionality. The newer computer, *Galicia*, had a number of working compilers installed, including FORTRAN and C++, along with supplementary software packages such as gmake. The computer also had its operating system upgraded to IRIX 6.5.2 to provide the best available kernel updates and hardware optimizations. Both *Breogán* and *Galicia* were connected to the SFU intranet and internet to facilitate data-sharing and enable remote management of the systems.

Molecular modeling research was performed. An insulin protein was simulated and studied in order to understand the principles of molecular modeling using CHARMM, MSI Insight II, and RasMol software.

IV. Discussion

Many software-related complications on *Galicia* forced time that was originally intended for molecular research to be repurposed to tackle the issues and leave *Galicia* in a much more improved and functional state than its condition before the research project began. The nature of the issues seems intrinsic to software and computer development in general: there are many variables that must be perfectly harmonious among multiple software packages and between all software and the underlying hardware. To be less abstract, software often has specific prerequisites that are implied or stated in brief – to the detriment

of a casual user or someone who is unfamiliar with the specific needs of the software.

The software-related complications can be addressed in primarily one way, that is, through improvements in user-friendliness. The IRIX operating system, many open source software packages, CHARMM, and LaTeX could all be 'tidied-up' to improve the user experience.

IRIX, as a member of the notoriously tricky-to-use yet powerful UNIX family of operating systems, left much to be desired in terms of allowing users to easily install new programs or effectively troubleshoot errors. Microsoft Windows, along with select other UNIX operating systems such as Ubuntu, have improved the user experience for software installation.

Open source software writers, such as the GNU group who produced the GCC and those who develop LaTeX, regularly make some effort to provide documentation of how their software works but often the writing style is in a technical format that can be cryptic to even adept users. On the contrary, some open source software, such as OpenOffice, has very user-oriented documentation that rivals or surpasses the quality of good commercial software documents.

CHARMM, despite its installation problems, was designed to be simple to install. The usage of this software was where complications arose, as the program was very powerful and focused in capability. The sheer number of simulation variables that could be tweaked or built was daunting to a newcomer to the molecular modeling discipline. Perhaps a simplified GUI could have served as 'training wheels' to prepare the new user for the complex software environment of digitized physical chemistry.

Since unresolved problems have indefinitely delayed the installation of CHARMM on *Galicia*, the computer may not yet be truly useful to a research project. Once the modeling program is functioning, the MSI Insight II program can be installed, which can serve as a visual front-end to CHARMM to aid in the manipulations of atoms, for example. Also, more hard drive space needs to be located for *Galicia*, via installing an additional internal SCSI drive, locating an external SCSI drive, or devising a network-based drive.

Alternative molecular modeling software exists in the event that CHARMM cannot be installed. Two QM-MM packages that may be utilizable are Amber and AMMP.

If no suitable researchers can be found who can carry on where the current researcher has left off, and if the cost of developing a molecular modeling computer is not of significant concern, perhaps funds could be allocated for purchasing a pre-configured computer system or for re-configuring an existing setup. A tech support contract may not be necessary since the computer is not likely to malfunction if it is used strictly for modeling purposes.

V. Summary

Molecular modeling is more than atomic simulations. The procedure and achievements that must occur before that of any modeling are as intensive, if not more so, than the modeling itself. Time, equipment, and research must be devoted to the process of developing a QM-MM computer. Over a period of almost two years, a pair of computers were 'poked and prodded' to see why they would not perform, until research and experimentation led to many solutions for both systems. A method of booting *Breogán* was devised, as was a method for installing special compact disk content on *Galicia* to update its operating system from IRIX 6.5 to IRIX 6.5.2. This update enabled the proper functioning of the GNU Compiler Collection software. The molecular modeling software CHARMM was not installed due to a misconfiguration on *Galicia*. Using a properly-configured computer from Montana State University, molecular research was performed on an insulin protein to exemplify the abilities of QM-MM. Finally, in writing this report, LaTeX was initially used but it was deemed unnecessary for meeting the typographical demands of this paper.

Further work can be done to install CHARMM, or an alternative modeling program, on *Galicia*. Until then, simulations of the molecular dynamics of α-conotoxin GI can perhaps be performed on *Zerner* at Montana State University. If research funds allow, a new version of CHARMM that features support for gfortran could be purchased and installed, as well as some mechanism for giving *Galicia* more free hard drive space to save research data.

VI. Footnotes

[1] Hargittai, Balazs and Pedro Muíño. NIH 1R15GM074654. "Role of disulfide bridges in the folding of conotoxins".

[2] Kochan, Stephen G. and Patrick H. Wood. *Exploring the UNIX System*. USA: Arcata Graphics Co. (1986).

[3] College of Engineering, University of Hawaii at Manoa. *Mastering the vi editor.* Fall 2006

[4] "MPFR 2.3.1". Fall 2006 <http://www.mpfr.org/mpfr-current/mpfr.html>

[5] Mpfr-2.1.1.tar.gz <http://www.mpfr.org/mpfr-current/mpfr-2.2.1.tar.gz>

[6] bootstrap-pkgsrc-IRIX64-6.5-mips-20040912.tar.gz <http://tinyurl.com/2d8zrj>

[7] "Linux@DUKE: Yum: Yellow dog Updater, Modified". Fall 2006 <http://linux.duke.edu/projects/yum/>

[8]CHARMM changelog for version c33 <http://www.charmm.org/package/changelogs/c33log.shtml#SECTION4>

[9]User 'rmv' statement of g77 compatibility <http://tinyurl.com/2el5b2>

[10]Forum post by 'Oneboy' <http://tinyurl.com/ynwlax>

[11] "RasMol Home Page". RasMol. Fall 2007 <http://www.umass.edu/microbio/rasmol/>

[12]Research Collaboratory for Structural Bioinformatics Protein Data Bank <http://www.rcsb.org/pdb/home/home.do>

[13] "proTeXt – a TeX distribution for Windows". Spring 2008 <http://www.tug.org/protext/>

[14] "MonkeyTeX: Online LaTeX Editor". Spring 2008 <http://www.monkeytex.com/>