**Gordon Chalmers – Software**

To address the request for more additional information about software, and in the context of not found or easily on the CV, this letter summarizes software packages that I have written and used in published works and also several software that have not been made public or used in published work. This is a serious position and I want to be taken seriously – I apologize for the length. It is my hope that this document will bring more easily the awareness to my experience and knowledge in computational software (my own and not my own) and molecular work.

I broke the discussion into 2 parts: 1st what is not known or available from my CV and 2nd , a brief summary of what is publicly available. The organized listing and very brief summary of these published packages in the 2nd make it easier for the committee to know, rather than looking at 3 different academic groups’ websites listed on my CV or the repositories on my GitHub site (http://github.com/GordonChalmers).

The unpublished works listed are all completed projects including software and uses. All have partial 1st drafts written and some have been described in the **GitHub: GRCPresentations** repository in the form of presentations that I gave. These works could also be used as starting points for a junior researcher investigating a molecular chemistry problem, such as the improved fit between measured IC50 assay data and computational prediction based on docking and protein-ligand interactions.

When asked about “local changes to a distributed scientific computing package that may have been used in a publication,” I have 3 comments. I have become acquainted with the use of GitHub in a group development of software from working in the Woods’ Glycam development team at the UGA CCRC, which I am not an active developer in now. In the past I have had to use LabArchives at RPI. Although not local, I have found 2 separate bugs in Matlab and Mathworks has repaired them, and 1 bug in CCDC Cluster GOLD (presumably corrected). One in Matlab was particularly nasty – I could send someone a heatmap created in the Bioinformatics toolbox in <2017a and ask to open it in a version >=2017a (heatmap format had changed then); it’s impossible to prevent the Matlab session from crashing after trying to open even after canceling. Again not local, I found a problem in the CCDC Cluster GOLD pre-docking batch preparation script cluster\_batch.py when used with .mol2 files of the input ligands to be docked. The input list of files that are batched into groups for use in Cluster GOLD are scrambled with 1 in BatchSize nulled from the input, and the mapping file has all file names changed, so that there is no connection with the original input; this can only be found by checking explicitly the mapping of input to output. Presumably this is fixed. Those aware should prepare their own batches for Cluster Gold if .mol2 ligand files are used. I do contribute to improving distributed software.

Regarding “Experience with lower level compiled languages such as C, C++ and Fortran is of particular interest,” I have programmed in C++ and C quite a lot. I have not used Fortran for years, but that was my first scientific language and am very familiar with it. I am familiar with calling code in one language from code in another and do so when necessary, and I am familiar with the many scientific libraries in these languages.

The software work that is not available from my CV is listed first, then my ongoing work from the last 2 years some of which is not available from CV. The last part is a brief synopsis of computational software developed at UGA CCRC.

***Unpublished work - software at Venenum***

My work at this company was in 2 projects. Their work follows a traditional hit search and then trial and error hit-to-lead optimization by iterative synthesis of compounds. I began with finding the protein target structures to do protein-ligand binding work, one a cryo-em structure from 2021 and the other from AlphaFold2. After that I spent my time with the head synthetic chemist on protein-ligand docking calculations and the use of these in selections of compounds to synthesize. All of this is proprietary, but one result from extensive docking calculations is pertinent to the public sector and is not known nor published:

*Docking implementation and improved IC50 calculations, statistics*

While at Venenum I had access to different types of in vitro assay data on molecules in development. Before joining Venenum I submitted a paper describing a more thorough use of docking calculations in explaining protein-ligand interactions; this recent paper **[1]** paper has been accepted for publication in the Journal of Computational Chemistry and uses a numerically calculated modeled energy density of states. Free energy of a protein-ligand complex (including entropy) is directly related to the value of an IC50 value measured in an appropriate assay through <E(T)>=coeff1\*ln(IC50)+coeff2, where <E(T)> is the Boltzmann temperature weighted expected value of E given a statistical density of states. I tested this relation following the recipe in the paper with 40 potential therapeutic ligands, all in the same molecular series and share substructure, on a target protein and found that the Pearson coefficient is .8 between the calculated and measured IC50 with an RMSD of .5. In conventional docking studies, only a highest scoring pose is used to quantify and the Pearson coefficient of the same set was .2 with an RMSD of 1.2. This larger scale test of the computational formalism in ‘Dynamic docking …’ is not in the paper, but the reason for the improvement is thoroughly explained. Although the molecules used can not be discussed, the analysis can be redone with any publicly available dataset of IC50 values. This is a significant improvement in the use of docking software in the practical developments of therapeutics using structural changes, synthesis, and assay testing.

**Unpublished work software at RPI:**

There are 3 software packages and work done at Rensselaer Polytechnic Institute that I would like to describe. All of the software is available for download and there is a PPT presentation with examples on each topic in the GRCPresentations repository in the GitHub site.

*Residual dipolar coupling (RDC) software and database*

While at Rensselaer Polytechnic Institute, amongst other tasks, I worked extensively on 2 databases. One was the design containing information of 150 small proteins (<150 amino acids) that used residual dipolar couplings (RDCs) in different media (phage, peg, …). I wrote software that computed the RDCs from the protein pdb files in a very systematic input/output Matlab structure format. This program gives almost the same output as REDCAT, but written in Matlab and is computationally fast. A different software performed statistical analysis between the back calculated and measured RDCs; this analysis also in a figure form can be used to further analyze the quality and accuracy of a protein structure that includes any number of RDCs. A paper was written explaining the analysis of this 150 protein database especially in regards to different definitions of ‘well-defined residues’. The latter is directly connected to degrees of flexibility in protein regions, i.e., dis-ordered or ordered. The paper is available on my GitHub site, and although unsubmitted due to the conclusions, the RDC software is available and simple to use with proteins and scalable with a for loop over any set (GitHub repository: **GitHub:** **RDC\_Software**). The RDC database is also available (not on GitHub) and is described in the paper.

*Molecule correlation time calculation software*

In an another project at RPI, I wrote a program that computes the rotational correlation time of individual proteins from measured T1 and T2 relaxation rates. These 2 quantities are typically used together in T1/T2 in a mass formula to obtain a 0th order estimate. The relation of T1/T2 and rotational correlation time was found heuristically years ago. However, a more accurate and fundamental spin pair approximation can be used instead, without regards to mass, because T1 and T2 are given in terms of the spectral density function at 3 values of \omega\tau. The ratio is a transcendental equation that can be solved numerically with a least squares fit algorithm (e.g., in Matlab) to find the correlation time of an amino acid in the protein if individual T1 and T2 values are known for the amino acid. The calculations were used in **[2]** but the method and software was not released. The difference between the heuristic formula and the spin-pair derived in terms of the spectral density can be up to 20% different. There is a series of papers in the 90’s that does approximate the solution to this transcendental equation for correlation time to a truncated cubic.

*Curve fitting with measurement uncertainty software*

Curve fitting a function to a series of data points, while including measurement error in the fit, is typically not done even semi-rigorously. I wrote software that does for a 1-D curve fit to a set of data points using an exponential weighting of the curve through the data point with error bars (std). It was planned for use in fitting 1-D spectral NMR data to find the errors on the fitted T1 and T2 values. The approach is new and can be used for any data series, 1-D and in higher dimensions. (This algorithm gives a unique answer, robust in testing, to the curve fit and is an alternative to the NMR RELAX software, which uses large amounts of random fits and an unclear fitness goal resulting in common crashes.) There is a presentation in the **GitHub: GRCPresentations** repository and the program is available.

**Unpublished work at UGA CCRC pre-graduation:**

There are several molecular software packages at UGA that much time was spent time. Two of these are the following:

*Glycoprotein structure improvement*

Deposited xray glycoprotein pdb files sometimes have bad coordinates chosen for atoms (xray experiments do not detect protons and have to be put in by hand). A genetic algorithm was written to reorient the xi,chi angles of rigidly attached ligands and thus the directions of attachment to minimize the overall clash. Gp120 is glycoprotein with 89 mid-sized carbohydrates attached (12 rings each) and the program successfully resolved all clashes. Oliver Grant (Research Scientist in Woods group) has a similar program using a Monte Carlo algorithm program, which runs faster, and we collaborated on this while the work was done. A presentation is given in the **GitHub: GRCPresentations**.

*Further NMR experiment modeling*

This is the only work mentioned in this document that is not completed; I did so because it is relevant to protein-ligand NMR work and in the category of modeling NMR observables as previous work. The modeling of saturation transfer difference (STD) from protein, ligand, and protein-ligand complex pdb files is a partially completed project. Spin diffusion and the on/off protein-coefficients are incorporated along with the methyl range saturation.

**Current software and work:**

Ligand (MultiProtein) GA is a computational molecular design tool, written in Matlab **[3]**. It consists of a genetic algorithm with non-isomeric SMILES based chromosomes to perform structure based drug design. The fitness function is that of protein-ligand interaction found from docking software. The first paper is to appear in Nature Scientific Reports. Details can be found in the paper and also in the **GitHub: Ligand\_GA\_version\_1\_1**, where the software can be downloaded. There is now an AutoDock version also, as opposed to the use of CCDC GOLD.

*Further: unpublished*

Since the initial paper submission Ligand GA has been generalized to Ligand Multi-Protein GA and coded to use a multi-protein system: high binding/interaction of a small molecule to a target protein, low binding/interaction to a set of others, or any combination. The original Ligand GA objective function can also now use a multi-objective function with only 1 protein: one function to maximize is the overall interaction score from docking and the second to maximize the interaction score per atom, the latter enforces specificity to the protein cavity against other protein cavities. Both of these together are a significant improvement. Fragment based design has also been included with a structural constraint on a portion of the initial molecule population, applying Ligand GA to build ideal fragments to of a total molecule for potency and biochemical activity. In addition, there were efficiency improvements in the calculations and memory handling. I have an incomplete draft of the generalization of Ligand GA to Multi-Protein.

The Ligand (Multi-Protein) GA software has been used on a variety of protein targets. It has been used extensively with published inhibitor results on the main protease Mpro of SARS-Cov-2 and Cox-2 **[4,5].** There is a write-up and extensive analysis of one of the generated inhibitors, not in any paper, to Mpro at the **GitHub: Recent\_Papers\_Work\_9\_22**; this molecule is potentially a covalent binder and is in a sense a descendent of Pfizer’s Nirmatrelvir (active ingredient in the drug Paxlovid=Nirmatrelvir+Ritonavir). The goal of the Multi-Protein use in this context is to increase the interaction with the Mpro active site and lessen the interaction to the CYP 3A4 enzyme, the latter reducing the metabolism rate and alleviating the need for an activator such as Ritonavir (which can cause multiple dangerous drug-drug interactions). In addition, I have selected 18 other non-covalent inhibitors that bind especially well to the Mpro active site and less so to the CYP 3A4 enzyme. **GitHub: Recent\_Papers\_Work\_9\_22**

In addition to SARS-Cov-2, the Ligand (Multi-Protein) GA software has been used on several other targets. In silico design of inhibitors generally can’t be published just about themselves without new methods or any synthetic work or in vitro testing. But I still enjoy working on producing in silico highly binding and specific protein inhibitors. First, I have results of its use to generate Hemagluttinin inhibitors to block Influenza A entry to the cell. Second, the bacteria pseudomonas syrangae has an ice nucleating protein (INAZ) on the inside of its cell well that raises the ice crystallization temperature from the typical roughly -40 C to -2 C via its surface shape and hydrogen bonding to water; this bacteria is the dominant mechanism of frost generation that destroys crops and other plants. It is being used to generate inhibitors of the inaz protein to block ice crystallization. Last, it has been used in proprietary work on a protein site that I won’t discuss, and on another target that is questionable for me to discuss as I never worked on it while at the company, but rather before joining and never assigned to or discussed this defunct (for >2 years) company project.

**Catalog of published software:**

These are software packages written in Matlab and/or C++ used for computational NMR work using molecular dynamics. The software can be used to interpret NMR experimental data and also to improve molecular modeling including the force field. They have all been used in published works and are available at either the Prestegard’s software site (http://tesla.ccrc.uga.edu/software/) or the Woods’ glycam site (http://legacy.glycam.org/docs/othertoolsservice/publication-related-materials/browse-publication-related-materials/index.html), and also at my GitHub site. The following are very short descriptions and in each download there is a complete documentation and examples. The downloads at Glycam have exceeded 1900, unique IP per package and total; the number from Glycam is listed in parenthesis and the statistics from the Prestegard software site is not collected (N/A).

**Assign\_SLP** (Glycam: redirect to Assign\_SLP\_1.12 at Prestegard software site, 444 and N/A)

is a genetic algorithm software package that finds the best assignment of experimental NMR spectral data to a protein. It uses NOE’s, chemical shifts, and rdc’s from measurements and the same from prediction to match and find the best assignment of NMR data to the small protein/molecule structure. This is generalized for larger molecules in Assign\_SLP\_MD. In both, missing measurements are allowed and taken into account in the assignments. **[6,7]**

**Assign\_SLP\_MD\_1.0** (Prestegard software site, N/A)

generalized this software package very much by using the trajectory in all aspects of the calculation, instead of only a single pdb frame as in its predecessor Assign\_SLP. This software uses the output of MD2NOE\_Protein, includes order parameters to improve RDC calculations, and uses a trajectory for the calculation of average chemical shifts. Statistical analysis and heatmaps are also generated which gives likelihood estimates of correctness for the individual peak assignments. **[8]**

**MD2NOE** (Glycam, 553)

calculates NOE build-up curves and NOE’s of small molecules, including carbohydrates, from an MD trajectory. It uses spin-diffusion of a many spin correlated system in the 2-spin pair approximation and takes into account spin relaxation globally throughout the many spin correlated system. It is a better approximation as result than that of the isolated spin pair (ISPA) which depends only on the 1/r^6 distance. A trajectory and topology file are used. One of the first packages of NMRBox. **[9]**

**MD2NOE\_Protein\_1.12** (Prestegard software site, N/A)

is a software package that calculates an NOE peak list, NOE buildup curves, and other relaxation parameters of large proteins from an MD trajectory. This package is large, and the output can be used in the genetic algorithm package Assign\_SLP\_MD. The package can use an arbitrary set of initial and final spin states of protons of large protons, which means that a general NMR experiment can be modeled. It is designed to model the sampling of motion found in MD trajectories from larger proteins by using a sphere approximated to effectively extend the time of simulation. The C++ code is memory use and computationally efficient (real-time speed) optimized and can use multiple CPUs. The output is a mimic of that of a spectrometer for a given NMR experiment. **[8]**

**Particular\_Relaxation\_Rate** (Glycam, 239)

calculates R1 and R2 relaxation rates of large and small molecules from an MD trajectory. It can be used for small or large molecules, such as large proteins. As in MD2NOE\_Protein, it uses the sphere approximation to enhance the sampling problem of not long enough trajectories for larger molecules. **[10]**

**3J\_Coupling\_Distribution** (Glycam, 430)

calculates 3J couplings from a molecular dynamics Amber trajectory and generates histograms for phi-psi plots of the conformations of molecules. The software is useful for structural calculations of carbohydrates, small molecules, and also protein-ligand complexes. It requires an .md trajectory file and topology file. **[10]**

**Single\_Frame\_NMR** (Glycam, 258)

calculates NMR observables: NOE’s using the isolated spin-pair approximation and 3J-couplings. These calculations use a pdb file or single frame of an MD trajectory. This program uses a pdb and topology file and has an organized output.

**In pure Computer Science:**

My computer science dissertation **GitHub: GRCPresentations** strongly centered on genetic algorithms and other techniques to approximate solution(s) to NP-Hard problems. The topic of scheduling and sensitivity analysis of real-time systems was the focus in the pure computer science side.

**Genetic\_Sensitivity**

is a software that uses a genetic algorithm to find an optimal parameterization of a fixed prioritized scheduling of cpus, where optimal is maximum utilization of the cpu resource. If the schedule is unfeasible (i.e., unschedulable) then it applies changes to task periods or deadlines in a fixed-priority task set. Unlike previous methods used in scheduling processors and examining sensitivity analysis, this software considers the parameters of all tasks as a whole globally by considering all the tasks in the task set simultaneously in a genetic algorithm. **[11]**

**References:**

**[1]** Gordon Chalmers, Dynamic docking in protein-ligand modelling. Accepted, in revision at Journal of Computational Chemistry. chemRxiv: 10.26434/chemrxiv-2022-6m0q5.

**[2]** Sriram Aiyer, G.V.T. Swapna, Li-Chung Ma, Gaohua Liu, Jingzhou Hao, Gordon Chalmers, Brian C. Jacobs, Gaetano T. Montelione, Monica J. Roth. A common binding motif in the ET domain of BRD3 forms polymorphic structural interfaces with host and viral proteins, Structure 2021 Aug 5, 29(8):886-898. bioRxiv 2020.09.21.306696. doi: 10.1101/2020.09.21.306696.

**[3]** Gordon Chalmers. Ligand GA: a genetic algorithm for automated protein inhibitor design. Accepted to Nature Scientific Reports. bioRxiv: doi: 10.1101/2021.10.11.463970. Software package:github/gordonchalmers/Ligand\_GA\_version\_1\_1

**[4]** Gordon Chalmers, Computational molecular modelling of Paxlovid binding. Submitted to Journal of European Medicinal Chemistry. chemRxiv: 10.26434/chemrxiv-2022-vfn6t.

**[5]** Gordon Chalmers. Computational Study of Paxlovid in Ligand GA. Submitted to Journal of Computer-Aided Molecular Design. chemRxiv: doi: 10.26434/chemrxiv-20220p2phq. Found also at, with supplementary information:

<https://www.dropbox.com/sh/3vuza1s0gjmy1eo/AADLXPPe2Ve-N3RjMH3GoH5fa?dl=0>

**[6]** Qi Gao,Gordon R. Chalmers, Kelley W. Moremen, James H. Prestegard. NMR assignments of sparsely labeled proteins using a genetic algorithm. Journal of Biomolecular NMR, 2017 April, 67:283–294. doi: 10.1007/s10858-017-0101-1.

**[7]** Kari Pederson, Gordon R. Chalmers, Qi Gao, Daniel Elnatan, Theresa A. Ramelot, Li-Chung Ma, Gaetano T. Montelione, Michael A. Kennedy, David A. Agard, James H. Prestegard. NMR characterization of HtpG, the E. coli Hsp90, using sparse labeling with 13C-methyl alanine. J Biomol NMR, 2017 Jul, 68(3):225-236. doi:10.1007/s10858-017-0123-8. PMID: 28653216. PMCID: PMC5546222.

**[8]** Gordon Chalmers, Alexander Eletsky, Laura Morris, Jeong-Yeh Yang, Fang Tian, Robert J. Woods, Kelley W. Moremen and James H. Prestegard. NMR resonance assignment methodology: characterizing large sparsely labeled glycoproteins, J Mol Biol. 2019 May 3, 431(12):2369-2382. doi: 10.1016/j.jmb.2019.04.029. PMID: 31034888. PMCID: PMC6554063.

**[9]** G. Chalmers, J.N. Glushka, Bethany Foley, R.J. Woods, J.H. Prestegard. Direct NOE simulation from long MD trajectories. Journal of Magnetic Resonance, 2016 Jan 21, 265:1-9. doi: 10.1016/j.jmr.2016.01.006. PMID: 26826977. PMCID: PMC4818662.

**[10]** Sheikh, Mohammed, Thieker, David, Chalmers, Gordon, Schafer, Christopher, Ishihara, Mayumi, Azadi, Parastoo, J. Woods, Robert, N. Glushka, John, Bendiak, Brad, H. Prestegard, James, M. West, Christopher. O2 sensing associated glycosylation exposes the F-box combining site of the Dictyostelium Skp1 subunit in E3 ubiquitin ligases, Journal of Biological Chemistry 2017 Nov 17, 292(46):18897-18915. doi:10.1074/jbc.M117.809160. PMID: 28928219. PMCID: PMC5704474.

**[11]** Gordon Chalmers, Shelby H. Funk, Adjusting Real-Time Mode Transitions via Genetic Algorithms, 2017 16th IEEE International Conference on Machine Learning and Applications. (presented at conference). Gordon Chalmers, Shelby H. Funk, Genetic Algorithms in Real-Time Systems, submitted to the Journal of Real-Time Systems, 2020.