Evaluating the Use of Spartan in Studying the Effects of Charged Lysine Residues

Mark Lazari

Abstract

Spartan by Wavefunction, Inc. is a powerful computational modeling tool that is used in both the research and academic realms.[1] This study reviews Spartan's ability to run conformational searches on alanine and lysine containing peptides, and compares Spartan's calculated values to those of Gaussian, another computational modeling program. The goal is to begin steps in understanding the effects of protonation of a lysine residue in a string of alanine residues, where lysine will reside either as the amino acid on the N-terminus or the C-terminus of the peptide. The study found that the protonation of a lysine residue placed at the C-terminus stabilized the α-helical structure of the peptide, and destabilization of the α-helix occurred when the protonated lysine residue was placed at the N-terminus. Spartan allows for a more user-friendly approach to conformational searches than Gaussian, and it gives a more graphical view of the conformations. This allows for better viewing of the structures and conformations while simultaneously calculating their thermodynamic data.

Background

Introduction to Computational Chemistry

Computational Chemistry has revolutionized our way of viewing molecules at the quantum mechanical scale by allowing us to simulate various chemical scenarios that are not possible to study in a lab. In Organic Chemistry, the transition states of molecules represent the theoretical structure molecules go through during a reaction. In lab, these transition states cannot be isolated. With Computational Chemistry, the transition states can be modeled, and their thermodynamic properties can be studied in various conditions (e.g. solvents, gas-phase, etc.) and at various temperatures. For Biochemistry, proteins can be better understood in terms of their behavior with other molecules (e.g. an enzyme binding to its substrate), and the structures and conformations of peptides and nucleic acid sequences can be more readily seen and understood via computational simulations. This *in silico* work draws its basis from what we have learned thus far from Physics and Physical Chemistry by bringing in wavefunction theory and various statistical mechanics methods based on quantum mechanics. The details of these theories and resulting mathematical methodologies will not be covered in this paper. For clarity, any mention of "methods" refers to the particular mathematical approach taken to run a calculation. Examples of these methods include but are not limited to semi-empirical AM1, molecular mechanics or MMFF, and density functional B3LYP/6-31G*).

As one might assume, something based solely on simulations from statistics and probability has its advantages and disadvantages. Many of the advantages are listed above, however, the disadvantage is clearly a lack of tangible, reproducible experimental lab results. Many *in silico* studies are done in tandem with experimental data, and many computational programs contain built-in protocols to help researchers better compare experimental data with the calculated theoretical results. By comparing reproducible computational data with reproducible lab data, one can better fine-tune the programming to continually gain more reliable results.

Spartan Review

One program that is used for computational analysis and molecular modeling is Spartan by Wavefunction, Inc. Spartan boasts a more user-friendly interface in terms of modeling and programming, which lowers the learning curve in both the academic and commercial realms of research. The modeling aspect is simple, straightforward, and allows for a wide range of molecules to be built quickly and efficiently. Spartan separates modeling "pieces" in terms of their relative uses, as seen in Figure 1 to the right, which is taken from Spartan '08. One tab in Spartan's build menu is set up specifically to deal with most organic molecule pieces, such as tetrahedral carbon atoms, trigonal pyramidal nitrogen atoms, as well as groups such as benzene rings and carbonyl groups. Another tab allows for more customizability, which is used most often for Inorganic Chemistry. This tab gives the user the ability to choose the atom, its geometry, and its bond type (i.e. single, double, triple, etc.). Spartan's builder also allows for amino acids and nucleic acids to be added individually or in sequence. For peptides, the user has the option to give the amino acid sequence an α-helix or β-sheet conformation as well as choose how to terminate the peptide at both the N-terminus and C-terminus. Using the peptide builder in concert with the other builder tabs, the user can easily adjust the peptide to fit his or her own personal needs.

Once the molecule is built, depending on the version of Spartan one has (e.g. '02, '06, '08), the user can view the molecule in various forms (e.g. ball and stick, tube, etc.), as well as reveal any hydrogen bonding, ribbons for proteins, labels, chiral centers, etc. Using some keyboard shortcuts and keyboard-mouse combinations, the user can also rotate the molecule at a chosen bond, adjust the bond length, and adjust the chirality of given chiral centers. For further customization, the user can lock certain bond lengths, angles, and dihedral angles, and use those constraints as starting points as well as limitations when optimizing the molecule's geometry. One example of this is forcing an ammonia (NH_3) molecule to remain in its trigonal planar geometry, which represents the transition state of ammonia during inversion. This can be seen in Figure 2 below. With these tools, the user can effectively build a molecule, view and adjust its restrictions and conditions, and label the molecule and its constituents for complete customizability all while having an easy to understand visual representation of these changes.

Figure 1: Build Tabs

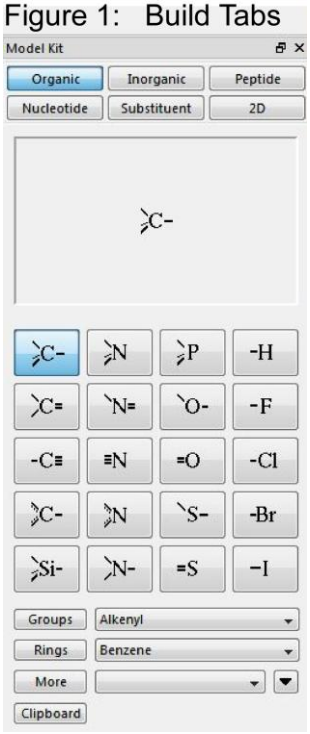
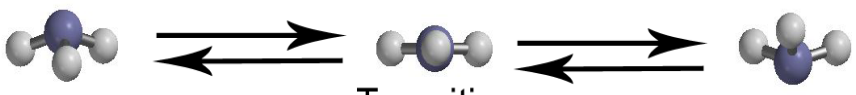


Figure 2: Inversion of NH_3



Transition State

Spartan Calculations

After a molecule has been built, a series of calculations can be done on it that will optimize it in terms of energy. Essentially, the calculations serve to find the lowest energy conformation, or the arrangement of the atoms of a molecule in three-dimensional space. How Spartan does this is dependent on the type of calculation desired, which are explain below. Spartan will also calculate a variety of chemical and physical properties that are common for all molecular modeling programs. The advantage Spartan has over many other modeling programs, however, is the fact that it can calculate and output a molecule's conformations at various energies. This is helpful in analyzing, understanding, and determining the physical representations of organic and biological molecules in both the energetically favorable (high energy) and unfavorable (low energy) conformations.

The calculations used in this study are conformer distribution, equilibrium geometry, and energy (or single point energy). Spartan offers other calculations for transition state analysis, other conformational search protocols, etc. that can be found in the program's handbook. These calculations are used to quickly, efficiently, and accurately determine the most energetically favorable conformations of each polypeptide. The methods are explained in the following sections, but the follow are brief summaries of what each calculation does.

Conformer distribution uses the Monte Carlo method of searching through possible conformations of molecules. The Monte Carlo method is described below. Spartan allows the user to customize the amount of conformations that will be searched through (e.g. a simple two residue peptide has on the order of 200,000 possible conformations) as well as tell Spartan to retain the structures of a given number of those conformations. What Spartan keeps is determined by the amount the user had defined to keep as well as the energy interval designated for the samples. By default, this interval is set to 40 kJ/mol, and only versions of Spartan '06 and above have the ability to adjust the default energy interval value. Spartan will search through and calculate the energies of all the conformations allowed by the user (e.g. 5000 of the 200,000) using the defined method (e.g. semi-empirical AM1) and generate in a separate file the number of conformations defined by the user, assuming it was able to find that many conformations within the defined energy interval. If 100 conformations are kept, Spartan will generate 100 conformations that are within an interval of energies that range from most energetically favorable to 40 kJ/mol higher than the favorable energy (assuming the default energy interval was used). Due to Spartan's graphical nature, these conformations can be easily viewed and judged both in terms of structure and energy. The amount of conformations the computer can search through and store are greatly limited by the computer's RAM capacity and available hard drive space. The speed, aside from the computer's processor quality, is determined by the method for energy calculations set for each conformation (e.g. semi-empirical, molecular mechanics, etc.).

Equilibrium geometry is one of the more straightforward calculations to run using Spartan, and its goal is to simply optimize the geometry of the molecule in terms of energy. What makes Spartan a powerful program is its ability to optimize a series of molecules stored within a given file. For example, if the 100 conformations retained from the conformer distribution calculation were determined using molecular mechanics calculations, and the user wanted to optimize each molecules' geometry in terms of AM1 calculations, the user could simply run each calculation the same as one would for a single

molecule. Spartan would calculate each molecule in the list one at a time until all molecules were optimized. This is called Global Calculations and can be seen in the calculations setup window. Spartan will also continue on to the next molecule if for any reason a molecule fails to be optimized, rather than cancel the entire run due to a single fault. An error would show up once the calculations are complete, and Spartan '08 will show these failed molecules in red in the spreadsheet. Otherwise, for older Spartan versions, the energy values would remain "pending" since the calculations were not completed for the failed molecules.

Energy or single point energy calculations are also straightforward to run in Spartan. They are often used for determining accurate thermodynamic data for each molecule. Often the more powerful methods of calculations are used for this (e.g. density functional B3LYP/6-31G*). Similar to the equilibrium geometry calculation, energy calculations can also be done for a series of molecules. This allows a user to build a peptide, gather 100 conformations within an interval of 40 kJ/mol out of 10,000 conformations searched randomly, optimize their geometries, and gather detailed energy values for each of 100 molecules with minimal setup. The time it would take to do this would only be limited by the computer handling the calculations and the method used rather than the user spending large amounts of time setting up the calculations.

Spartan also has the same breadth of methods and calculations that many other modeling programs have. These range from simple molecular mechanics calculations to a variety of semi-empirical and ab initio methods. The later versions of Spartan add more "point and click" option changes on the calculations screen, however, any custom changes can be programmed in using the Options bar. Here, options like wanting the output data to contain the three-dimensional (3-D) coordinates of the atoms of the molecule, adjusting the conformer distribution energy interval, adjusting temperature values and limitations, adding solvent conditions, etc. can be added or adjusted. Spartan files can hold many molecules, and often it is advantageous to append molecules to a single Spartan file in order to plot energy diagrams, build animations, as well as study and calculate values for multiple molecules more efficiently. Together with its graphical nature, Spartan offers high quality modeling that is intuitive, straightforward, and only limited in certain customization options found in other modeling programs.

Goal

This study aims to analyze the conformations of small polyalanine and single lysine containing peptides as the single lysine residue is protonated. We are studying the effects of the protonation of the lysine on the α-helical nature of peptides when the lysine is placed either on the N-terminus or the C-terminus of the peptide. It is believed that that an α-helical peptide forms a permanent dipole due to the arrangement of the amide formed in peptide bonding, where the N-terminus holds the partial positive charge and C-terminus holds the partial negative charge.[2] This study will test whether the position of the charged lysine residue will have any effect on the α-helical nature of the peptide.

Results from the calculations done with Spartan will also be compared to the results generated from calculations done with Gaussian, which is a comparable modeling program that allows for more custom calculation options but is considered less user-friendly (i.e. has a greater learning curve). Initial usage has shown that because of its graphical nature and greater automation, setting up calculations with Spartan appears to be more straightforward and take less time than the equivalent calculations done with Gaussian.

Materials and General Methods

Three versions of Spartan, by Wavefunction, Inc., were utilized for this study, because of the slight variations in their options and their availability in lab. They were Spartan '02, Spartan '06, and Spartan '08. Spartan '06 was used for the majority of the calculations because of its greater availability. The only differences that exist between the versions of Spartan are the amount of features available for the user and the accessibility of those features.

The computers used for the majority of the calculations were Dell Precision 690 workstations running Microsoft® Windows® XP 64-bit, an Intel® Xeon® 5140 processor @ 2.33GHz, and two PC2-5300 2GB RAM sticks @ 333MHz. Note also that all the calculations done in this study were for molecules in the gas-phase, which is a theoretical consideration. The use of gas-phase calculations eliminates the variability that arises when other molecules are near the peptide being studied. It is important to note that peptides do not naturally exist in the gas-phase.

Peptide Construction and Labels

Multiple residue peptides were tested in this study. The peptides were constructed using the Peptide tab under Spartan's build menu. Here, single amino acids are available as pre-built models to be added separately or to existing models. Also, a sequence of amino acids could be created and organized as α-helix, β-sheet, or some custom formation. For this study, all molecules were constructed with initial α-helical conformations to calculate the energies of conformational changes caused by the protonation of the lysine residue. The sequence begins with the N-terminus and ends with the C-terminus, and each terminus can be protonated or deprotonated by using the terminate command in the build screen. Once the peptide sequence is built and added to the workspace, the N-terminus was N-acetylated ($-C=OCH_3$) and the C-terminus had an OH group added to make it a carboxylic acid. The reason for these additions is to limit the ability of the peptide to be protonated at another site other than at the lysine residue. All peptides, unless otherwise stated, have the N-terminus acetylated and the C-terminus as a carboxylic acid, and the order in naming and organizing the peptides will be N-terminus to C-terminus (e.g. AK is a two residue peptide with alanine at the N-terminus that is acetylated and a lysine at the C-terminus as a carboxylic acid). Since the study aims to see the conformational changes resulting from the protonation of the lysine residue, both the protonated and deprotonated forms of the peptides were built. To distinguish between the two, an H was added at the end of the label (e.g. for the AK peptide, AKH would be the protonated version). These labels will be used throughout this report, though this form of labeling is not conventional.

Results, Discussion and Detailed Methods

Conformer Distribution

Once a peptide was properly built (see Peptide Construction and Labels), the peptide was put through the conformer distribution calculation. The calculation methods used were both molecular mechanics (MMFF) and semi-empirical (AM1) methods. The test was to see what would be the most efficient method of testing through the Monte Carlo method, which is described below.

For the MMFF method, the calculations take far less time, and thus more conformers can be searched through in a given time interval. The molecule of choice for the testing was the four residue peptide A3KH (i.e. A3 corresponds to three consecutive alanine residues). The conditions of the calculation were these:

CONFSEXAMINED = 10000 CONFSKEPT = 50 MAXENERGY = 5 PRINTCOORDS

with CONFSEXAMINED and CONFSKEPT as options available on the calculation screen for Spartan '06 and above, representing the amount of conformations to search through, and how many to keep, respectively. MAXENERGY refers to the energy interval to search through (see Spartan Calculations in the Background section, and note that Spartan '02 does not have this option), and the PRINTCOORDS simply displays each atom's 3-D coordinates in the output. The peptide was subject to symmetry in order to begin conformation variation from the initial α-helix built (see Peptide Construction and Labels above), and the thermodynamic data was printed as well.

For the AM1 method, the above conditions were identical except the method was set to semi-empirical AM1 rather than molecular mechanics MMFF. On the computers used in this study, the MMFF calculation took between 3-5 hours to complete, while the AM1 calculations took almost a week. Due to the nature of the calculation methods, the energy values were not able to be directly compared, therefore, the MMFF conformations displayed were also run with AM1 equilibrium geometry in order to run the extra AM1 calculations the AM1 conformer distribution underwent. To remain consistent, the AM1 calculation also had an AM1 equilibrium geometry calculated on it, though it had little to no effect, which was expected.

The reason the AM1 calculation took much longer lies within the fact that each of the 10,000 conformations that were searched through had their geometries optimized with AM1, whereas for the MMFF method, only the 50 conformations that were kept had to be optimized with AM1 in order to be comparable to those initially run with AM1. Initially running MMFF proved to be the most efficient method, and the results of the most energetically favorable conformations from each test were within five significant figures in kcal/mol (Table 1 below shows a different molecule as an example). However, the conformations that resulted from each method were different, which is in the nature of the Monte Carlo method. Analysis of conformation vs. energy is discussed below in the Helical Nature vs. Energy section.

To better study the differences in methods, a systematic approach is necessary that is considered reproducible. This is in contrast to the Monte Carlo method, which is not reproducible when using a large data set. The Monte Carlo method allows a random sample to be tested, thus giving the user a greater chance of avoiding being caught in local energy wells, which is a common issue with the systematic approach. The systematic approach allows for a predetermined sequence of calculations to be run in order for a reproducible result to be obtained from any consistent starting point (see Spartan Handbook). The systematic approach was not tested at this stage, though Spartan does have the ability to run such a protocol.

MaxEnergy

Extensive tests that did not use the MAXENERGY option for the conformer distribution calculation revealed that the program does indeed give the user an interval of conformation energies equal to 40 kJ/mol. When the MAXENERGY option was used to limit the value to 5, the program seemed to have adjusted the range to 5 kcal/mol instead. The reason behind has not yet been found, but it could have something to do with the program deciding what units would be best for the given molecule. It also could have been a coincidence, and other tests have shown that Spartan seemingly ignored the maxenergy command (see Helical Nature vs. Energy below and compare energy ranges). More tests need to be run with Spartan ’06 or above, which have the MAXENERGY option.

Comparison to Gaussian

The reliability of Spartan was tested against another modeling program, Gaussian, which was used and verified by Dr. Jianhua Ren's research team at the University of the Pacific in California. Gaussian '03 was compared to Spartan '06 in regards to energy calculations of the conformations that were determined from Spartan '06. The peptides tested were AK and AKH, both of which had 20 conformations kept from a pool of 3,000 conformers searched using the conformer distribution calculation with the AM1 method. The 20 conformers' geometries were optimized using AM1, and the energies were calculated using the density functional B3LYP/6-31+G* method. The optimization and energy calculation aspects were done separately using each of the programs to determine an energy value for each conformer in Hartrees, where one Hartree is equal to 627.509 kcal/mol. Below is a list of these energies for comparison:

Table 1: Energy Comparison of Gaussian '03 and Spartan '06

	Gaussian '03		Energy in Hartrees	Spartan '06	
Molecule	AK	AKH		AK	AKH
1	-897.035	-897.430		-897.036	-897.430
2	-897.024	-897.416		-897.024	-897.425
3	-897.015	-897.420		-897.015	-897.416
4	-897.021	-897.420		-897.021	-897.420
5	-897.018	-897.418		-897.018	-897.426
6	-897.028	-897.418		-897.028	-897.423
7	-897.029	-897.426		-897.029	-897.425
8	-897.024	-897.421		-897.024	-897.413
9	-897.017	-897.420		-897.017	-897.413
10	-897.027	-897.411		-897.027	-897.418
11	-897.020	-897.419		-897.021	-897.418
12	-897.033	-897.427		-897.033	-897.441
13	-897.019	-897.409		-897.019	-897.420
14	-897.027	-897.438		-897.027	-897.435
15	-897.028	-897.432		-897.028	-897.432
16	-897.030	-897.426		-897.030	-897.432
17	-897.041	-897.428		-897.041	-897.431
18	-897.025	-897.430		-897.025	-897.428
19	-897.037	-897.430		-897.037	-897.430
20	-897.030	-897.425		-897.031	-897.428

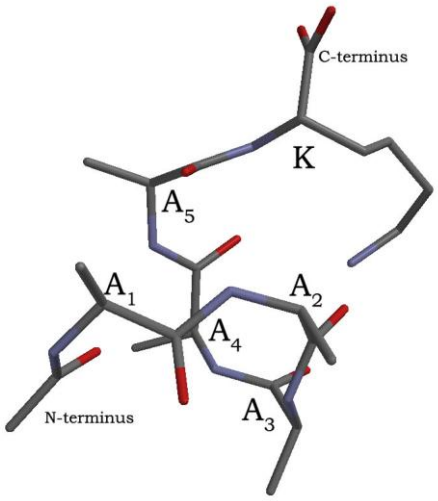
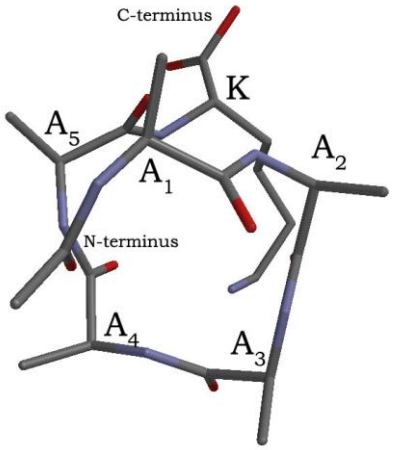
Table 1. Energy was calculated using B3LYP/6-31+G* after optimization with AM1 from each of the respective programs. The initial 20 molecules came directly from a conformer distribution calculation using AM1 from Spartan '06 of both a protonated and deprotonated lysine of an alanine and lysine containing peptide that was acetylated on the N-terminus and had a carboxylic acid as the C-terminus.

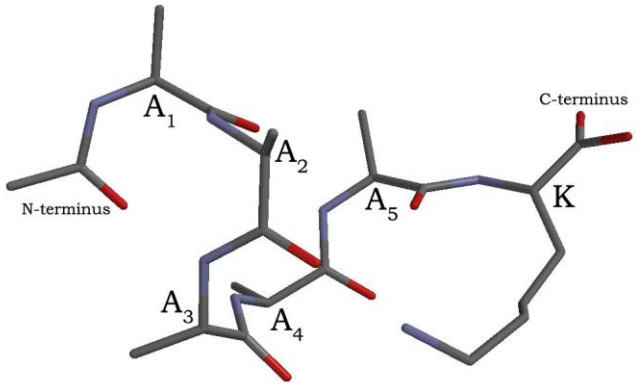
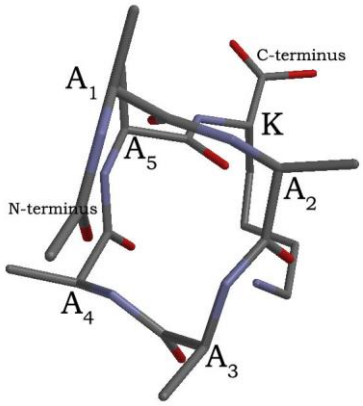
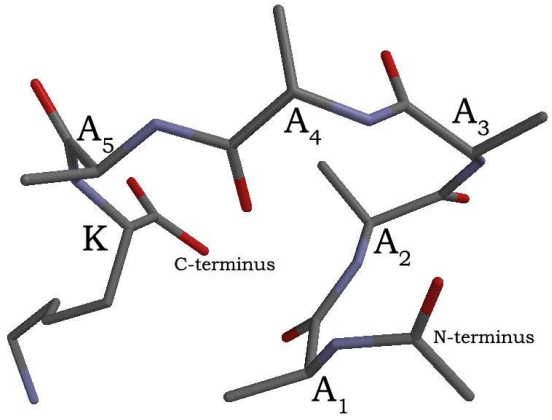
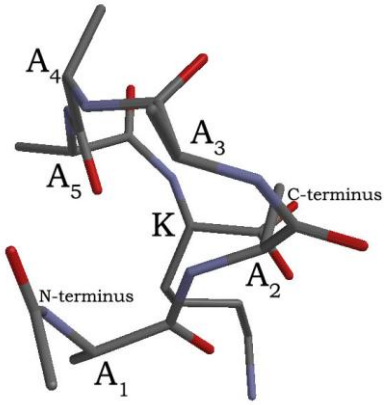
From the above table, the differences in energies between the values obtained from Gaussian and Spartan occur at the fifth or sixth significant digit. The reason for these changes must be in the subtleties within the calculation process and programming rather than the actual equations used in the methods. This allows these programs to be used in tandem with each other based on the needs and preferences of the user. For example, Spartan can be used to run the conformational search while Gaussian can be used to optimize them, or Spartan can optimize and Gaussian can do further analysis with custom calculations not available in Spartan.

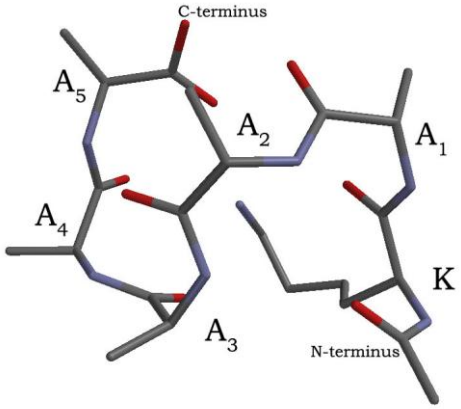
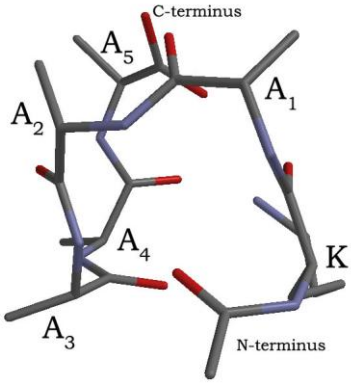
Helical Nature vs. Energy

Through various conformational searches, it seemed somewhat evident that the lower (i.e. more negative) energy does correlate to a helical structure of a peptide. Since all these studies are in the gas-phase, the conformation, orientation, intra-molecular forces, etc. of the peptides determine its relative energy, rather than interactions with solvent molecules or other intermolecular forces. Though peptides occur in α-helical and β-sheet conformations in protein folding, in the gas phase, it is not necessarily certain that the lowest energy conformation must be α-helical, let alone β-sheet. Below are six conformations. Three come from A5KH and three from KA5H (see Peptide Construction and Labels for naming rules). The three conformations consist of the two most energetically favorable and the least energetically favorable when 10,000 conformers were searched through using conformer distribution using MMFF. Spartan kept 200 conformations, which were then optimized using AM1 equilibrium geometry. The MAXENERGY command was also set to 100, though it seemed not to have any affect.

Table 2: **Helical Nature vs. Energy Comparison**

Molecule	Energy (kcal/mol)	Side View	Through the Center View
Most energetically favorable	-256.702		

Next energetically favorable	-256.339		
Most energetically unfavorable	-194.603		

Most energetically favorable	-250.250		

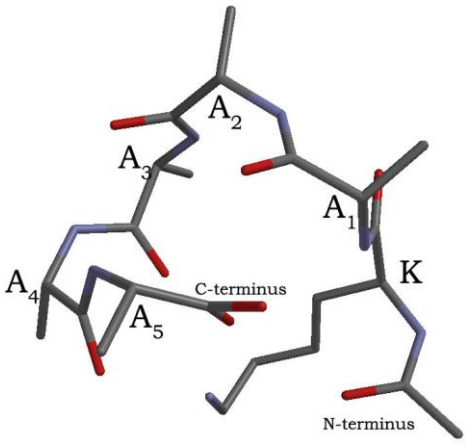
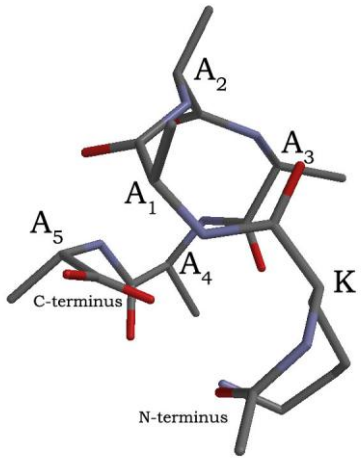
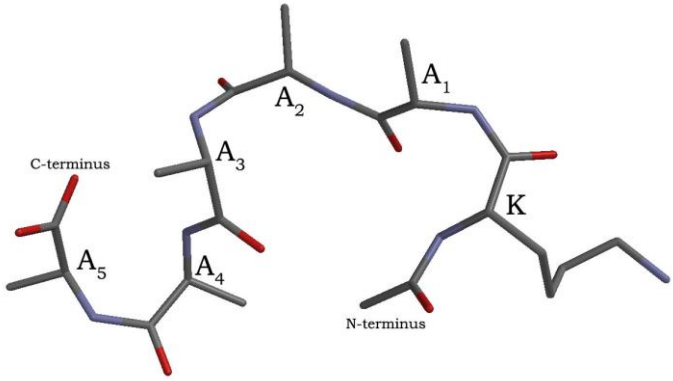
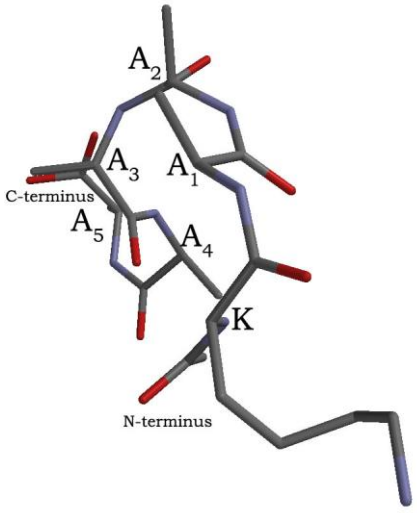
Next energetically favorable	-249.979		
Most energetically unfavorable	-191.793		

Table 2. *The hydrogen atoms have been removed and the style of model is tube in order to show the structure with the least amount of clutter while still maintaining a clear view of the molecule. Two different views are offered for each peptide as well as labels to help identify the location of the amino acids as well as the N-terminus and C-terminus. Purple is nitrogen, red oxygen, grey carbon, and double bonds and possible hydrogen bonding are not shown.*

From this sample, it is clear to see that the A5KH, which has the charged lysine residue placed at the negatively charged end of a possible permanent dipole, maintains somewhat of a helical shape when viewing the most energetically favorable conformations. The most energetically unfavorable of the A5KH conformations searched through does not form a helix.

KA5H seems to lose its helical shape regardless of energy. KA5H does show that the two energetically favorable conformations are quite different in shape, though similar in energy. The most energetically unfavorable KA5H conformation shows a more unraveled peptide, which results in a loss of any form of helical nature.

The results here are not conclusive by any means, but serve more as a stepping stone for further research on these peptides. There appears to be evidence that placing a positively charged species near the negative end of the permanent dipole seems to stabilize the helical nature of a peptide, whereas placing the same charged species near the positive end on the permanent dipole seems to disrupt the peptide's helical nature. Also, from the KA5H, there seems to be evidence pointing towards a discrepancy in assuming that energetically similar conformations would also be similar in shape. This seems true for A5KH, which appears to be most stable in the helical nature, thus the most energetically favorable conformations would be suited best in the helical form. This is not true for KA5H, which seems disrupted by the position of the charged lysine residue.

Issues and Errors

RAM

RAM stands for random access memory, and it is used to store information that will be sent to the central processing unit (CPU) for calculation. Spartan requires more RAM as more conformations are needed to be searched through. Preliminary tests were run with only 1GB of RAM in each of the computers, and this proved to be quite difficult when larger peptides needed to be calculated. Spartan shows the error of having insufficient memory to continue the calculations. To remedy this, the computers were upgraded to 4GB of RAM each, and this allowed for more conformations to be run for larger peptides.

Spartan Crashes

Occasionally when running calculations on sets of peptides (e.g. after a conformational search), Spartan will finish the calculation and yet show an error message upon completion. What this often means is that Spartan was not able to finish the calculations on all of the molecules, but it was still able to complete others in the set. If no particular issue was found for the failed peptides, sometimes rerunning the calculation solved the problem. Some things to check are charges and multiplicities, as well as any constraints that may not be applicable to all of the molecules in the set. This study was not able to determine all the reasons for the errors.

Monte Carlo Method

The Monte Carlo method is a powerful tool used to run the conformer distribution calculation. This method uses a statistical mechanics approach to randomly sort through potential energies. For conformational searches, the Monte Carlo method allows the user to search a population of conformations at random, in order to gain a more uniform look at the sample as a whole. With systematic methods, predefined patterns are used to move a molecule through potentials, so as to create a reproducible approach to searching through conformations. The Monte Carlo method uses the previous conformation's potential as it searches, to determine whether or not to keep the value or reject it. In terms of the conformational search, it seems as though the method takes the potential of the previous conformation, puts it into an equation if the potential is higher than the previous, and compares the output to a randomly chosen number, and in this way the program randomly keeps potentials that are not always moving towards a lower energy, in order to avoid getting caught in local minima energy wells. According to Atkins and De Paula, $0 < e^{-\Delta V/kT} < 1$, where ΔV is the change in potential energy (which is positive if the next potential in the series is higher than the previous), k is Boltzmann's constant, and T is the temperature.[3] The method must then generate a random number between 0 and 1, and compare the value obtained with the random number. Due to this randomization, local energy wells can be avoided, but the results are not reproducible or necessarily reliable to

be the lowest energy. Therefore, it is recommended that the Monte Carlo method be repeated to in order to obtain a accurately expressive set of conformations.

Conclusion

Spartan is a powerful computational program that allows for a variety of calculations to be done, including the valuable conformational search. The energy calculation values match with those of Gaussian, another computational program that can offer more detailed calculations and protocols, but Gaussian is not as user-friendly as Spartan.

If the alanine and lysine helical peptides do form a permanent dipole where the N-terminus holds the positive end and the C-terminus hold the negative end, then putting a protonated (i.e. positively charged) lysine residue at the C-terminus seems to stabilize the helix whereas placing the charged lysine at the N-terminus seems to disrupt the helix. More calculations and examples need to be performed to test this hypothesis, but this study has shown that helical stability can be seen and calculated using Spartan.

The next step would be to study the proton affinity of the lysine residue as well as test more conformations to see how the charged species affect the helical stability of these peptides.

Acknowledgements

Special thanks to the Chemistry Departments at both the University of the Pacific and California State University, Stanislaus for providing the computers and software necessary to perform this study, and to Dr. Jianhua Ren under whose auspices this study was possible, and to Dr. Scott Russell for providing for the RAM.

References

(1) Spartan. *Wavefunction, Inc.* 2008. http://www.wavefun.com/index.html

(2) Oommachen, S., Ren, J., and Michael McCallum, C. Stabilizing Helical Polyalanine Peptides with Negative Polarity or Charge: Capping with Cysteine. *J. Phys. Chem*. 2008. *112*, 5702-5709.

(3) Atkins, P., DePaula, J. *Physical Chemistry*. 8th Edition. W.H. Freeman and Company, 2006. Page 607.