

An Interactive Computer Program to Help Students Learn Molecular Symmetry Elements and Operations

David E. Meyer and Andrew L. Sargent*

Department of Chemistry
East Carolina University
Greenville, North Carolina 27858
Ph. 252-328-9759
Email: sargenta@ecu.edu

Abstract

We report a new, fully interactive 3D visualization computer program designed to help students learn molecular symmetry elements and operations. The program allows students to define symmetry elements in essentially any location within a molecule and effect the corresponding symmetry operation. In a separate display window, the result of the operation is superimposed with the original molecule image in order to facilitate a comparison of the two, and to determine whether the symmetry element, as defined, is present within the molecule. Written in Java and platform independent, the program executable, a User Guide, and a library of thirty-six molecules with interesting symmetry are available for download from the author's faculty web page at <http://www.ecu.edu>. Additional molecules can be read into the program, provided that simple format guidelines are followed.

Keywords: Symmetry and Group Theory, Computer-Based Learning, Upper-Division Undergraduate, Inorganic Chemistry

An Interactive Computer Program to Help Students Learn Molecular Symmetry Elements and Operations

David E. Meyer and Andrew L. Sargent*

Department of Chemistry
East Carolina University
Greenville, North Carolina 27858
Ph. 252-328-9759
Email: sargenta@ecu.edu

Introduction

A thorough understanding of molecular symmetry and group theory is an integral part of the undergraduate curriculum in chemistry. Typically, the subject is taught as part of a junior-level inorganic chemistry course, but its applications span all four traditional disciplines - analytical, organic, inorganic and physical - and include numerous key topics of biochemistry. The pedagogical approach is universal: introduce the concepts of molecular symmetry elements and operations, explain how to follow a decision tree based on the presence or absence of said elements in a given molecule to arrive at the molecular point group, and apply this knowledge to applications such as optical activity, orbital interactions, spectroscopy, and chemical reactivity.

The problem is that the steps of the pedagogical approach require the student to mentally visualize the molecule in three dimensions, manipulate the 3D image in very specific and often subtle ways, and compare the result of the manipulation to the original image. Not all students are adept at this type of mental visualization. To aggravate the problem, the material used to teach the process is routinely presented through 2D media. Modern textbooks employ fancy drawings replete with wedged and dashed bonds, specular color, and intricate shading to

simulate three-dimensionality. Yet, despite this effort, many students still have trouble “seeing” these 2D representations in 3D. To facilitate the process, the conscientious instructor brings a ball-and-stick model kit to lecture. This, however, presents its own set of problems. First, the models are difficult to manipulate. In order to demonstrate a reflection operation, for example, atoms must be plucked off each side of the mirror plane and placed back in the appropriate locations on the opposite sides. This is cumbersome, at best, and is often accompanied by hand waving and “Imagine this” phrases in an attempt to authenticate the mediocre visual effect. Further, what students see in the instructor’s model depends on where they are seated in the classroom. The threefold rotation axis in an octahedral model, for example, is difficult to see unless the student is looking directly down the axis. In order for all of the class to benefit from the experience, the model kit must be moved and carefully oriented to offer each student the prime perspective, which is impractical in most cases.

Computer visualization software solves many of these problems by providing a simple point-and-click windowing environment to facilitate the manipulation of 3D images, thereby eliminating the need for complex mental visualization. Several software programs have already been reported which take advantage of computer visualization tools to aid the instruction of molecular symmetry. Most can be categorized as tutorials (1-2), multimedia presentations (3-4), or computer-assisted instruction (5-6) and possess a modest degree of user interactivity. Others appear to be more interactive, but are difficult to categorize due to a lack of current availability (7). Impressive animations are becoming routine as educators incorporate sophisticated web tools such as the Chime (8) and Shockwave (9) browser plugins, or Java applets (10). One contribution (9) is particularly noteworthy in that it possesses impressive Direct X-rendered graphics, a robust library of 47 molecules of interesting symmetry, and significant user

interactivity. All of the symmetry elements of each of the forty-seven molecules are available for display, with impressive Shockwave animation showing the corresponding symmetry operation. Although the use of this program can contribute significantly to the preliminary stages of teaching and learning group theory, the student should ultimately have the opportunity to define the symmetry element wherever they want, in any molecule of their choice, and be able to effect the corresponding symmetry operation to see the consequence of their choice – correct or incorrect. This is the motivation behind our efforts.

Herein, we report a new visualization program, SymmetryApp[©], which utilizes a different level of user interactivity, heretofore absent in programs reported to date, to the best of our knowledge. In combination with textbook and classroom instruction, SymmetryApp is an effective tool for helping students learn the complex details of symmetry elements and operations. The key to its success is three-fold. First, the program is platform independent and easy to install. SymmetryApp is written in the Java programming language and can run on any computer platform from Mac and Windows to Linux, requiring only a simple installation of the Java Runtime Environment available free from <http://www.java.com>. Second, SymmetryApp is capable of reading any Cartesian coordinate file following the .xyz file format (*vide infra*). While a library of thirty-six symmetrically significant molecules is included, the program is by no means limited to only these choices. Third, and most importantly, SymmetryApp exploits the benefits of interactive learning; no predetermined flow of information is prescribed the way it is in a tutorial or slideshow presentation. Students have complete control within the program to define symmetry elements in virtually any location in a molecule, and effect the corresponding symmetry operation – whether or not it leads to a “correct” answer. Exploration of this type is practically limitless, since the user can define and execute a large number of different operations

within a single molecule. Such a program allows students to learn at their own pace and, importantly, have the opportunity to make mistakes – and learn from them.

Program Design

In discreet chemical systems, five unique classes of symmetry elements exist and may be present in various combinations within a particular molecule. A symmetry operation involves a specific, well-defined mathematical transformation about a symmetry element, and results in a second image of the molecule. If the second image can be identically superimposed on the original image, the molecule is said to contain or possess that symmetry element. Determining the specific number and type of symmetry elements that yield such superimposable images is a prerequisite to assigning the point group that uniquely classifies the molecule according to symmetry. SymmetryApp is designed to allow the user to define symmetry elements within a molecule, calculate and display the effect of the corresponding symmetry operations, and facilitate a comparison of the two molecular structures via simple superposition of the molecule images and by a comparison of the corresponding Cartesian coordinates. Algorithms exploiting standard linear algebra techniques are used to perform the mathematical operations (11).

SymmetryApp is based upon the framework of another program available under the GNU General Public License. MolStart (12), a basic molecular viewer and editor available as freeware from <http://molstart.sourceforge.net>, was coded in Java 1.4.2_03 by Chengteh Lee and Libero J. Bartolotti. While MolStart has undergone several revisions in its history, an older, more streamlined version was used as the framework for SymmetryApp. Most of the molecule-editing features were stripped from the program and several of the methods were rewritten to provide the functionality required by SymmetryApp. However, the main window of the program, which

includes the molecule visualization frame, and all of the code for rendering a molecule in three dimensions, remains effectively unchanged.

At its outset, SymmetryApp executes a graphical user interface that allows the user to visualize and manipulate a molecule in three dimensions using a mouse. The main method of the program opens a window frame with three major components. The first component, a panel that holds the three-dimensional representation of a molecule, is placed in the center of the frame. The second component consists of both a message bar and toolbar that reside below the visualization panel at the bottom of the window. The message bar allows for dynamic instructions to be displayed when the user is defining symmetry elements. It displays error messages when the user makes a mistake, and queries the user if further clarification is needed. Beneath the message bar is a toolbar that contains the usual radio buttons to rotate, translate, and magnify the perspective of the molecule. It also contains two action buttons, *Reset* and *Go!*, each of which becomes active at appropriate times during the course of the program's execution.

The third component, a menu bar, appears at the top of the window. The *File* menu allows the user to open molecule files from either the included library (*vide infra*) or some other source, close the current molecule, open new SymmetryApp windows, or save the Cartesian coordinates of the molecule to a new file. The *Edit* menu contains two choices: *Undo* and *Measure*. The former serves to remove the most recently defined dummy atom (see *Define* menu below) while the latter allows the user to calculate bond distances, measure bond angles and dihedral angles, and create or delete bonds. The *Define* menu provides a way to create dummy atoms as points in space that may be useful or necessary in defining a particular symmetry element. Standard selections include a *Midpoint*, which creates a dummy atom midway between two user-selected atoms, and a *Centroid*, which places one at the mathematical center of a user-defined ring of

atoms. Additional selections include *Perpendicular To A Line* and *Perpendicular To A Ring* that are similar to *Midpoint* and *Centroid*, but differ in that the dummy atom is offset from the line or a ring plane. Without dummy atoms, certain symmetry elements of some molecules would be impossible to define using only the molecule's constituent atoms. Lastly, all five of the symmetry elements are listed as selections in the *Symmetry* menu. In most cases, selecting a symmetry element from the menu opens a new window that contains a series of options specific to that element.

Once the symmetry element has been defined, the user can execute the corresponding symmetry operation. A new visualization window is automatically opened and two images of the molecule are displayed: those corresponding to the molecule before and after the symmetry operation is performed. If the specific symmetry element is indeed present as defined by the user, the two images will be identically superimposable and will thus appear indistinguishable from the image of the molecule in the original visualization panel. However, if the symmetry element is not present, the two molecules will not be identically superimposable, and will appear, instead, as a sort of double exposure. In either case, the user gets clear visual feedback regarding the validity of their choice. As a secondary check, the program automatically compares the Cartesian coordinates of both molecule images and displays information regarding the comparison to the user in a message bar.

Program Execution: An Example

To illustrate a few key features of our program, we will examine the chair conformation of cyclohexane, which is included in our library of molecules, and step through the execution of a C_2 symmetry operation.

Double-clicking the SymmetryApp executable (a Java Jar file) begins execution of the program. Selecting *Open* from the *File* menu allows the user to browse the local file structure to locate the downloadable library that contains the molecule of interest. The molecule can be freely rotated with simple trackball point-and-click operations in order to view the 3D structure from any perspective. Since the atoms of this example molecule do not reside in locations that would allow the user to define the principal rotation axis, dummy atoms need to be defined. The *Define* menu presents several choices – we select *Centroid*. Once this selection is made, dynamic messages appear in red type within the message bar below the visualization panel to guide the user through the necessary steps. As we begin to select atoms, the atom labels are listed in the message bar to help the user keep track of the atoms selected. When the selection is completed, the *Go!* button in the bottom right corner of the toolbar becomes active and, when clicked, effects the placement of the centroid within the molecule. Although the second point necessary to define the rotation axis could be defined by a second centroid of another ring of atoms in this molecule (such as either set of three alternate carbons around the ring, or by either set of three axial hydrogens) we'll choose a different route to illustrate another useful program feature. At the bottom of the *Define* menu are selections to define dummy atoms perpendicular to a line or ring. By selecting *Perpendicular To A Ring* and clicking on a ring of atoms in a plane orthogonal to the rotation axis (in this example, either set of alternating carbons around the ring), we achieve the desired result. The options of the *Symmetry* menu allow us to select the type of symmetry element we wish to define with the atoms present in our window. Following the selection of a (Proper) Rotation Axis, a secondary window pops up with options. Dynamic messaging appears in various locations within this window to help us select our rotation axis options. We select C_2 to specify the order of the rotation axis, and messages in blue text at the

top of the window instruct us to select one point (of two) to define this axis. We select the dummy atom that was defined as a perpendicular to a ring. Red text confirms our selection and

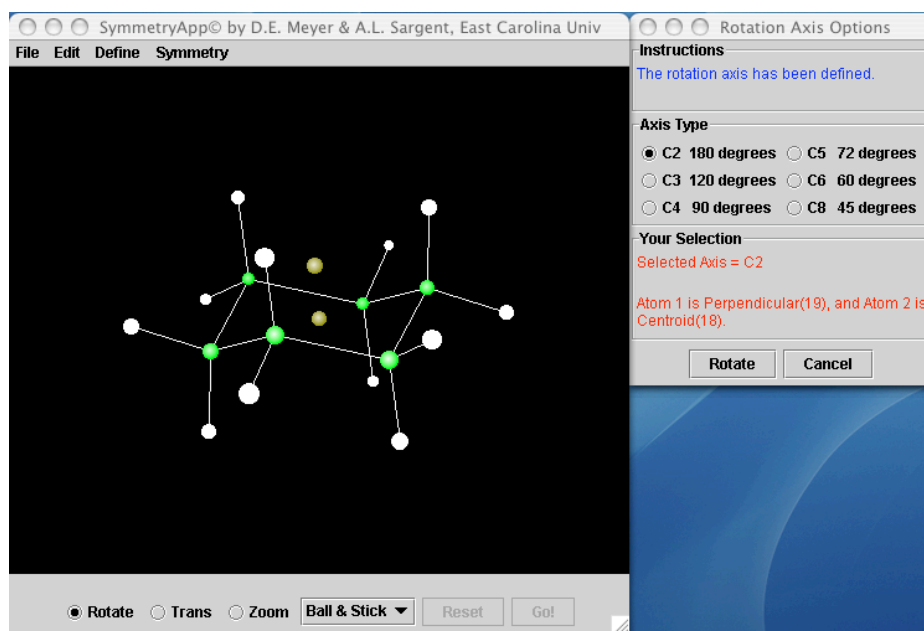


Figure 1. The chair conformation of cyclohexane is displayed in the main visualization panel, along with two user-defined dummy atoms (gold) to define the proper rotation axis: the centroid and perpendicular to a ring. Dynamic messaging appears within the Rotation Axis Options window to assist the user in defining the symmetry element. Here, the axis has been defined and the Rotate button has been activated.

the blue text instructs us to choose a second atom on the rotation axis. We select the centroid, thereby completing the definition of our symmetry element and activating the *Rotate* button in the options window, as shown in Figure 1. When we click *Rotate*, a second visualization panel appears, shown in Figure 2, wherein two molecule images are displayed: the original molecule and that which results from the C_2 symmetry operation. The user is free to rotate the superposition of molecules in the new window to view it from any perspective. It is clear from the superposition that the C_2 symmetry element is not present in the molecule as we defined it in this example – the molecule is not rendered indistinguishable from the original image. Red text

at the bottom of the superposition window provides additional information that results from a comparison of the Cartesian coordinates before and after the symmetry operation is effected.

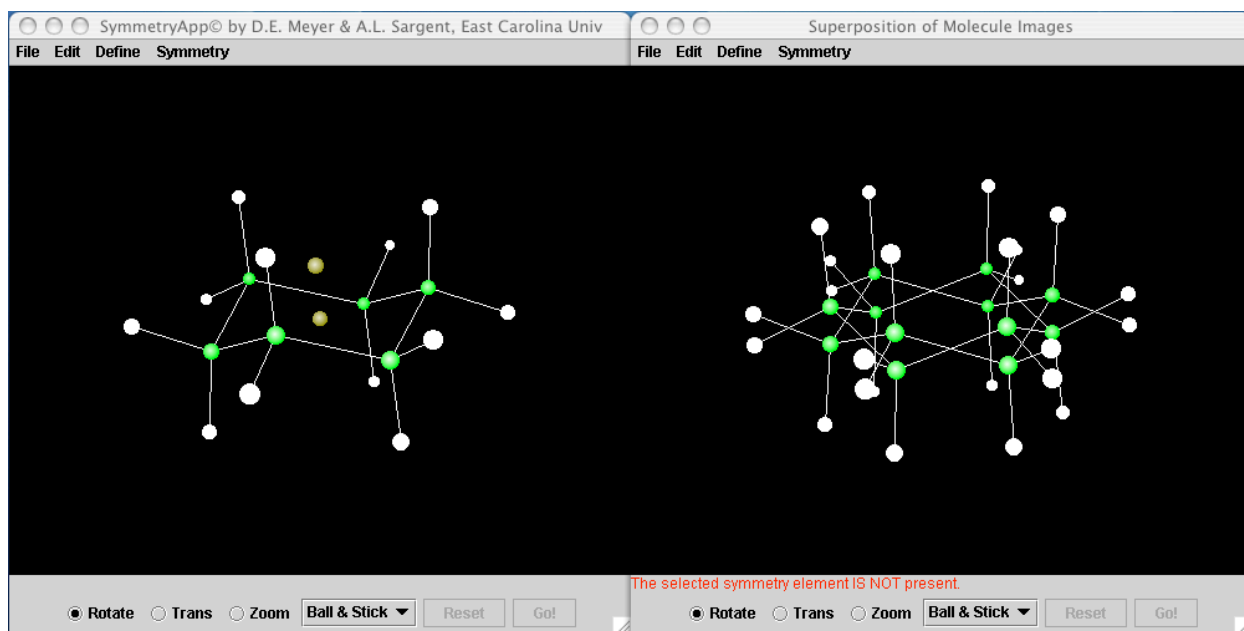


Figure 2. A second visualization window, containing a superposition of the molecule images before and after the symmetry operation has been executed, appears when a symmetry operation is executed. Red text at the bottom of the superposition window confirms, from a comparison of the Cartesian coordinates of the two molecule images, that the C_2 symmetry element is not present as defined in this example.

In this case, the text confirms that the C_2 element is not present as defined. Moving our cursor back to the *Symmetry* menu of the primary window and selecting a new symmetry element results in the disappearance of the superposition window and the appearance of the options window described earlier. If we select a C_3 proper rotation axis and define it with the same two dummy atoms, once the operation is effected we see what appears to be just a single molecule image in the superposition window when, in fact, it contains both images perfectly and indistinguishably superimposed. The text at the bottom of the window confirms what we see from the graphics window – the C_3 symmetry element, as defined, is present in the cyclohexane molecule.

Similar steps are required to define and execute mirror planes, inversion centers, and improper rotation axes. Care has been taken to ensure that the program contains sufficient flexibility to allow the user to easily define symmetry elements in nearly any location – regardless of their validity. For example, when defining an inversion center, the associated options window provides the user with two choices – to select either the center of molecular mass (COM) or some user-defined point, which can be any of the atoms in the molecule or any dummy atom. In a molecule such as the chair conformation of cyclohexane, choosing the COM would yield an indistinguishable superposition of images, but so would the selection of the centroid we defined from the six carbon atoms. The experienced eye recognizes that this centroid coincides with the COM, but the novice student has the opportunity to discover this on his or her own. If the student selects a point that does not coincide with the actual center of inversion, the program faithfully executes the corresponding symmetry operation and renders the result graphically, providing clear visual feedback.

Another example of the flexibility built into the program concerns the improper rotation axis. If the user selects a rotation axis that possesses the COM, the program automatically makes the sensible choice to reflect through a plane that contains the COM. If the rotation axis does not contain the COM, the program prompts the user to select a point that resides in the reflection plane. From an algorithmic standpoint, this is necessary in order for the collective rotation-reflection operation to be defined in a reliable manner. From the educational standpoint, this allows the user to define the improper rotation axis in virtually any location, regardless of the accuracy of the definition.

For linear molecules, defining a rotation axis orthogonal to the molecular axis requires the use of a specific set of dummy atoms. In particular, the *Midpoint* and *Perpendicular To A Line*

selections from the *Define* menu allow the user to easily generate the necessary dummy atoms. A valid criticism of the program is that it doesn't allow for an infinite number of choices for the order of rotation axes, such as can be applied to linear molecules. However, we made the judgment call that the simplicity associated with the finite number of choices presented in the Rotation Axis Options windows better serves the student. An earlier version of SymmetryApp included an advanced module where the user could type any order for a proper or improper rotation axis, and could also specify the number of consecutive operations by typing a superscript in the element notation (e.g. the "2" in C_3^2). Although the algorithm worked well, its presence in the program created more confusion than added functionality and it created clutter in an otherwise clean look. Ultimately, its presence didn't significantly add to the pedagogical aim. For linear molecules, the student can easily execute proper rotation operations of a number of orders, say, C_2 , C_3 , C_4 , C_5 , C_6 , C_8 , etc. (i.e. all of those available from the Rotation Axis Options window) and therefore infer that any order operation is valid.

Availability

The corresponding author's faculty web site at the East Carolina University Department of Chemistry provides instructions and a link to download a precompiled Jar file of SymmetryApp that needs only the Java runtime environment. Additionally, a guide to getting started with the program is available with simple instructions and screen captures illustrating key aspects of its operation. Finally, a zip file containing a library of thirty-six molecules with interesting symmetry is available to download. The molecules included in the library are simple (most are smaller than twenty atoms) and represent many of the common point groups. All of the files in the library follow a simple .xyz format, in which the first line contains the total number of atoms (Natom) in the molecule, the second a title line (which may be blank), and then Natom lines that

begin with either the atomic number or atomic symbol followed by the Cartesian coordinates, space delimited, all in free format. Although the library files are general documents with an .xyz file extension, SymmetryApp can also read text and Microsoft Word files with .txt and .doc file extensions, respectively, provided they adhere to the same general format. This added functionality allows the user to import molecules for inspection that lie outside the scope of the included library. Instructors wishing to tailor the program to fit a particular lecture or homework set need only to create the molecules of interest in a molecule builder and save the coordinates in the .xyz format. Alternatively, .xyz files can be downloaded easily from the Internet using any search engine.

Classroom Implementation and Assessment

SymmetryApp has been tested over three semesters in upper-level inorganic chemistry courses. Typically, the program is projected onto a screen at the front of the classroom and used to introduce symmetry elements. Such a presentation serves two purposes. First, it overcomes most of the pitfalls associated with the topic's presentation via 2D media or ball-and-stick model kits, as mentioned in the Introduction. Second, it provides a quick tutorial on the use of the program, so that when assignments are distributed, the students already have a general idea how to proceed.

A survey taken to assess student attitudes and perceptions of the program was taken during the spring semester of 2005. Students were asked to provide information regarding their use of the program and answer several questions by numerically ranking each from 1 (completely disagree) to 10 (completely agree). Table 1 lists the questions asked and summarizes the responses.

Table 1. Summary of survey results of student attitudes and perceptions regarding use of the SymmetryApp program. Responses are numerical and range from 1 (completely disagree with question) to 10 (completely agree). Class size was thirty-two students; population standard deviation is also listed.

Question	Average \pm Stdv
Hours of use	2.6 \pm 0.9
Number of times program was accessed	2.3 \pm 0.8
Program was easy to use	7.2 \pm 2.0
Program worked well	8.2 \pm 1.4
3-D model kits were used	4.2 \pm 3.4
3-D model kits are better	5.7 \pm 2.4
Textbook alone was adequate to learn symmetry	5.5 \pm 2.3
Want to have program available on the web	9.4 \pm 1.2
Can identify symmetry elements better after using program	7.0 \pm 1.9
Program had a lot of bugs	1.8 \pm 1.5
Program was helpful in lecture	8.3 \pm 2.0
Program was valuable in learning symmetry	7.6 \pm 1.9

The survey was taken when the program was available only on departmental lab computers, and most students indicated that their usage would increase with if they had access via the web. Overall, the students found the program to be helpful, not only outside of class, but in the lecture as well. Some still used standard 3-D model kits, but the general sentiment was that neither model kits nor the textbook (*13*) alone was sufficient for learning symmetry elements and operations.

In the comments section of the survey, students said they liked the freedom to work at their own pace, the opportunity to make choices (especially incorrect ones) and felt that the program did a good job of facilitating both, helping them learn about symmetry elements and their corresponding operations.

Acknowledgments

SymmetryApp was developed with support from the Center for Interdisciplinary Instructional Technology and Research at East Carolina University.

References

1. Pottillo, L. A.; Kantardjieff, K. A. A self-paced computer tutorial on the concepts of symmetry. *J. Chem. Educ.* **1995**, *72*, 399-400.
2. Korkmaz, A.; Harwood, W. S. Web-supported chemistry education: Design of an online tutorial for learning molecular symmetry. *J. Sci. Educ. Tech.* **2004**, *13*, 243-253.
3. Lee, A. W. M.; Leung, K. M.; Kwong, D. W. J.; Chan, C. L. Symmetry elements and operations. *J. Chem. Educ.* **1996**, *73*, 924-925.
4. Vining, W. J.; Grosso, R. P. Symmetry and point groups. *J. Chem. Educ.* **2003**, *80*, 110.
5. Kastner, M. E.; Leary, P.; Grieves, J.; DiMarco, K.; Braun, J. Point Group I, II, and III. *J. Chem. Educ.* **2000**, *77*, 1246-1247.
6. Nedwed, K.; Gatterer, K.; Paulson-Fritzer, H. SYMAPPS 1.0: A software packet for group theoretical applications to molecular symmetry. *Comput. Chem.* **1994**, *18*, 371-376.
7. Vázquez-Vidal, L. MOLSYM: A program on molecular symmetry and group theory. *J. Chem. Educ.* **1996**, *73*, 321-322.
8. Johnston, D. Symmetry tutorial. <http://www.otterbein.edu/home/fac/dnhjhns/symmetry/index.html> (accessed date July 2006).
9. Charistos, N. D.; Tsipis, C. A.; Sigalas, M. P. 3D molecular symmetry Shockwave: A web application for interactive visualization and three-dimensional perception of molecular symmetry. *J. Chem. Educ.* **2005**, *82*, 1741-1742.
10. Cass, M. E.; Rzepa, H. S.; Rzepa, D. R.; Williams, C. K. An animated interactive overview of molecular symmetry. *J. Chem. Educ.* **2005**, *82*, 1742-1733.
11. Meyer, D. E. Computational Tools in Chemistry: From Software Development for the Advancement of Chemistry Education to Classic Quantum Chemical Research. M.S. Thesis, East Carolina University, Greenville, NC, 2004.
12. Lee, C.; Bartolotti, L. J. MolStart. <http://www.molstart.com> (accessed July 2006).
13. a) Vincent, A. *Molecular Symmetry and Group Theory*, 2nd Ed.; John Wiley & Sons, Ltd.: England, 2001. b) Miessler, G. L.; Tarr, D. A. *Inorganic Chemistry*, 3rd Ed.; Pearson Prentice Hall: New Jersey, 2004.