IIT MADRAS

GROUP 12

Chemical Formula Visualizer

Author:
Aditya Sapate (CS10B031)
Kalyan Kumar (CS10BOO5)
Kodali Praveen (CS10B014)

Supervisor:
Dr. Sukhendu Das

May 5, 2014

Problem Statement:

Biomolecules, such as proteins and nucleic acids (DNA and RNA), are involved in every aspect of cellular function. Often times, understanding their structure is key to understanding their function. In the past, crystallographers and biologists created detailed real-world models, called Corey- Pauling-Koltun models, using wooden or synthetic spheres to represent atoms and sticks to represent bonds. Today, these models of protein structures, referred to as space-filling and ball-stick models, have been adopted in computer graphics systems to create visual representations.
Your job is as follows:
Input: A Chemical Formula eg. C6 H12 (Cyclohexane).
Output:

- Visualize it using ball-stick models. ☑
- Visualize it using Space filled models. ☑
- Your demo should produce smooth edges around the intersection of spheres and cylinders ☑
- control view angle at different zoom level. ☑

Packages Required:

- OpenGL/Glut
  # sudo apt-get install freeglut3 freeglut3-dev
- biniutils-gold (for the linker)
  # sudo apt-get install binutils-gold
- RDKIT (for translating .smi files to .mol files)
  # sudo apt-get install rdkit∗
- OPEN BABEL (for translating the code from .mol to .cml files
  # sudo apt-get install openbabel

Color Scheme:

1. Atoms
   - Oxygen (O) Red
   - Nitrogen (N) Blue
   - Phosphorous (P) Orange
   - Carbon (C) Grey
   - Hydrogen (H) Green
2. Bonds
   - Single Bond = Yellow
   - Double Bond = Magenta
   - Triple Bond = Cyan

Flow Structure:

Input: SMILES formula of a given chemical formula
Output: 3-D structral drawing of the chemical formula

SMILES formula $\mapsto$ MOL formula $\mapsto$ CML file $\mapsto$ 3-D structural mapping

Location of the files generated:

SMILES formula: Entered as input in the "converter.py" file.
.mol file: Stored in file "./data/temp.mol" using "rdkit package"
.cml file: Stored in "./output.cml" file using "obabel package"
3-D formula: Output is shown on the screen after compiling and running the code using "./run.sh"

Implementation:

Input compound conversion

We tried many ways to take molecules in IUPAC representation as input. Somehow we were not able to bridge the gap between IUPAC names and spatial representation of the molecules. So we went through various file formats to represent chemical compounds. .smi(SMILES) format was the one which seemed favourable as it could be converted to .mol format thus giving us the spatial locations of the atoms *(using rdkit)*.

Now the second challenge was that now we had the spatial location of molecules but no clue which molecules formed the bond. So we went a step further. We converted .mol format into .cml format .. thus getting both atomic locations and bonds amongst compounds *(using openbabel)*.

Initializations

Basic Initializations like creating a window, defining the functions for mouse and keyboard interrupts, setting illumination options like lighting, POV, etc .....

Parsing of .cml file

cml file is a sort of xml file which contains all the atomic location of elements and all the bond angles. We created 2 data structures to store all this vital information in the program itself

- **Bonds:** for storing all the values concerning bonds like single/double/triple bond, atom id amongst which bond is present.
- **Elements:** for storing all the location coordinates of each of the atoms with their id.

Ball-Stick Model

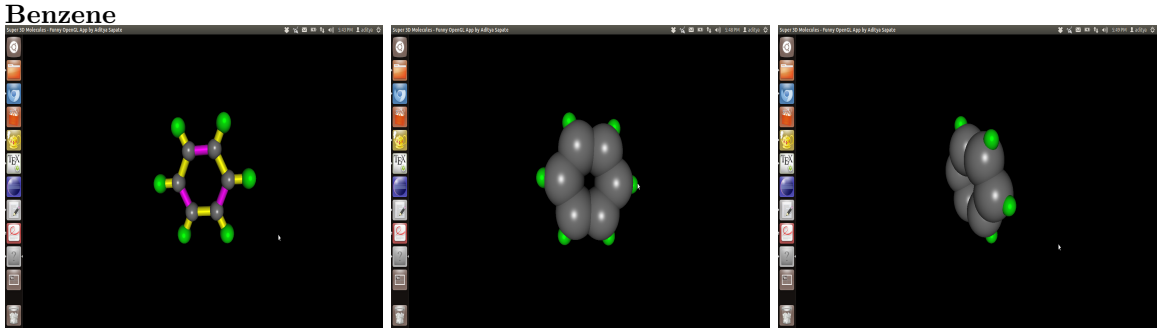
1. Draw the atoms at the corresponding location
2. Using bonds in the molecules, we connected the atoms using a cylinder.

Space-Filling Model

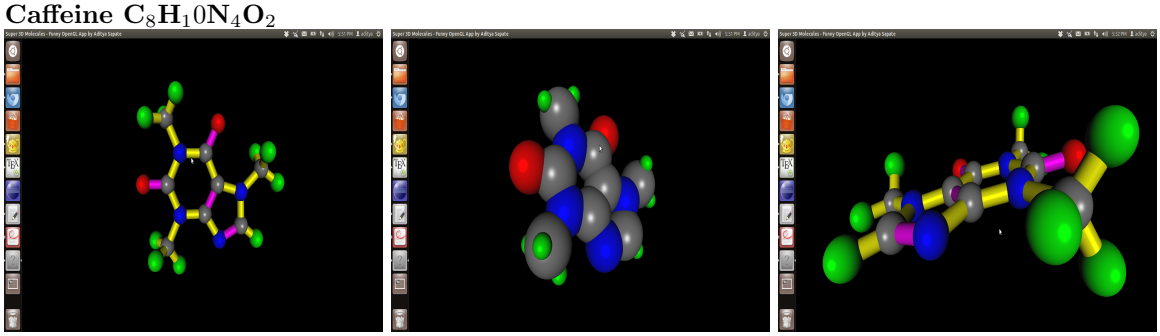
1. We used the radius of each atoms and cubed them as
   volume $\propto$ (radius)^3
2. Now as a reference we divided all the (radius)^3 by the value of (carbon_radius)^3 as a reference
3. We multiplied the quotient of this computation with a multiplier (determined by trial and error) for scaling.

Sample Outputs:

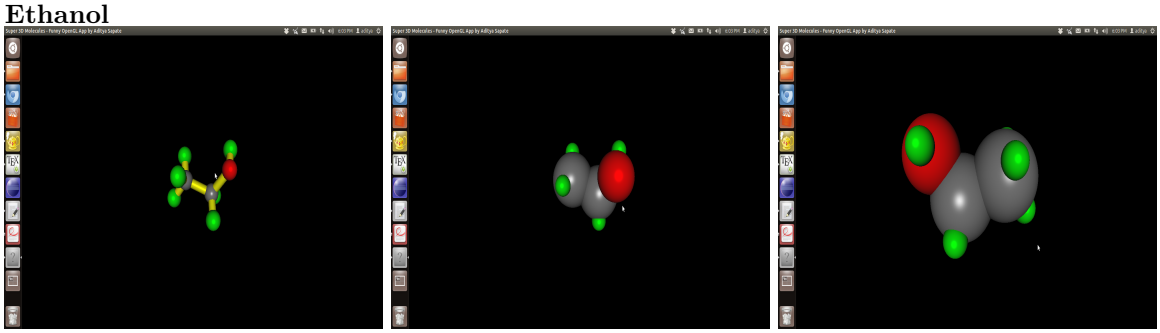
Benzene



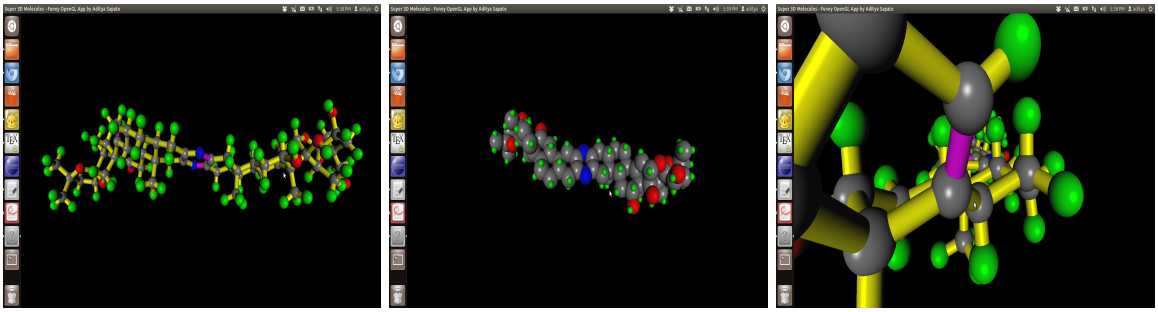
Caffeine $C_8H_10N_4O_2$



Ethanol



A molecule with more than 9 rings, Cephalostatin-1 ($C_54H_74N_2O_10$)



Abbreviations:

- **sml: Simplified Molecular Input Line Entry Specification (SMILES)**
  SMILES strings include connectivity but do not include 2D or 3D coordinates.
  Sample SMILES format

Name	Formula	SMILES string
Methane	CH_4	C
Ethanol	C_2H_6O	CCO
Benzene	C_6H_6	C1=CC=CC=C1 or c1ccccc1
Ethylene	C_2H_4	C=C

- **cml: Chemical Markup Language**
  Chemical Markup Language (CML) is an open standard for representing molecular and other chemical data.

- **mol: Molfile**
  An MDL Molfile is a file format created by MDL for holding information about the atoms, bonds, connectivity and coordinates of a molecule.

References:

http://www.opengl.org/documentation/
http://en.wikipedia.org/wiki/Chemical_file_format
http://rdkit.org/docs/
http://openbabel.org/docs/2.3.1/index.html
http://lifeofaprogrammergeek.blogspot.in/2008/07/rendering-cylinder-between-two-points.html