MASARYK
UNIVERSITY

FACULTY OF INFORMATICS

# Modern visualization of partial atomic charges in Mol*

Bachelor's Thesis

DOMINIK TICHÝ

Brno, Spring 2023

MASARYK UNIVERSITY

FACULTY OF INFORMATICS

# Modern visualization of partial atomic charges in Mol*

Bachelor's Thesis

DOMINIK TICHÝ

Advisor: RNDr. Tomáš Raček, Ph.D.

Department of Computer Systems and Communications

Brno, Spring 2023

FACULTAS ARTIS INFORMATICÆ • UNIVERSITAS MASARYKIANA •

## Declaration

I declare that I have worked on this thesis independently, using only the primary and secondary sources listed in the bibliography.

Dominik Tichý

**Advisor:** RNDr. Tomáš Raček, Ph.D.

# Acknowledgements

TODO

# Abstract

TODO

# Keywords

Molstar, molecular visualization, partial atomic charges

# Contents

# List of Tables

# List of Figures

# Introduction

# 1 Theory

Theoretical concepts are fundamental to the study of computational chemistry, providing a framework for analyzing molecular structures and properties. This chapter focuses on four key areas of theory, beginning with an overview of molecular structure in Section 1.1. Section 1.2 provides an in-depth examination of chemical file formats, including the advantages and disadvantages of different file formats for storing molecular data. Partial atomic charges are explored in section 1.3, including their importance in the analysis of molecular structures and the various methods used to compute them. Finally, section 1.4 delves into color interpolation, a critical technique for visualizing partial atomic charges in molecular structures.

By providing a comprehensive overview of these theoretical concepts, this chapter aims to provide a strong foundation for the subsequent chapters, which will focus on the implementation and analysis of the Molstar extension for visualizing partial atomic charges in molecular structures.

## 1.1 Molecular structure

Molecular structure refers to the arrangement of atoms and chemical bonds in a molecule. The three main components of molecular structure are atoms, residues, and chains. In this section we will look at

### 1.1.1 Atoms

Atoms are the basic building blocks of matter. Atoms are composed of protons, neutrons, and electrons. [citations]

Bonds are the connections between atoms that hold molecules together. There are different types of bonds, e.g. covalent bonds, ionic bonds, and hydrogen bonds. [citations]

### 1.1.2 Residues

Residues are the repeating units in a polymer chain. In proteins residues refer to amino acids, which are connected through peptide bonds to form polypeptide chains. [citation]

### 1.1.3 Chains

Polymer chains (chains) are sequences of residues that are linked together. In the context of biomolecules, chains can be either polypeptide chains in proteins or polynucleotide chains in nucleic acids. The sequence and structure of these chains are crucial for understanding the function and properties of the biomolecules. [citation]

## 1.2 Chemical file formats

TODO: introduction

### 1.2.1 SDF

Structure-data file (SDF) is a widely used chemical file format for representing molecular structures and their associated properties. It is a text-based format that describes the atoms, bonds, and atomic coordinates of a molecule. [citation]

### 1.2.2 MOL2

The Mol2 file format is another text-based format for storing molecular structures and their associated properties. It can store multiple conformations of a molecule and is commonly used in molecular modeling and cheminformatics applications. The Mol2 format provides more flexibility and additional features compared to the SDF format, such as support for multiple substructures and atom types. [citation]

### 1.2.3 PDB

The Protein Data Bank (PDB) file format is a widely used format for storing three-dimensional structures of proteins, nucleic acids, and other macromolecules. PDB files contain information about the atomic coordinates, secondary structure, and other important details required for understanding macromolecular structures. The PDB format has been widely adopted in structural biology, bioinformatics, and related fields. [citation]

### 1.2.4 mmCIF

The macromolecular Crystallographic Information File (mmCIF) format is an extension of the CIF format, specifically designed for macromolecular structures. It is a text-based format that provides a more comprehensive and flexible representation of macromolecular crystallography data compared to the PDB format. The mmCIF format supports data dictionaries, allowing users to define new data items and integrate additional information. Additionally, the format is not limited to a fixed amount of lines. [citation]

TODO: add example image + better explanation

## 1.3 Partial atomic charges

Partial atomic charges are a measure of the distribution of electronic charge within a molecule. These charges are important for understanding and predicting molecular interactions, including hydrogen bonding, electrostatic interactions, and solvation effects. [citation]

TODO: visual of electron distribution

### 1.3.1 Calculation methods

TODO

### 1.3.2 ChargeFW2

### 1.3.3 AlphaCharges

## 1.4 Color interpolation

Color interpolation is the process of creating new colors by mixing two or more colors together. It is a common technique used in computer graphics and digital image processing to create smooth transitions between colors.

Color interpolation works by calculating the intermediate colors between two or more given colors. This is typically done by taking a weighted average of the red, green, and blue values of the colors being interpolated.

TODO: add math equation + image of red,white and white,blue interpolation

# 2 Visualizing molecular data

This chapter will discuss various types of visualizations, the role of coloring in molecular representations, and some commonly used software tools for creating these visualizations.

## 2.1 Types of visualizations

There are several methods to represent molecular data, each with its own benefits and drawbacks. The methods most relevant to this work are the following three types: ball and stick, surface, and cartoon.

### 2.1.1 Ball and stick

The ball and stick model represents atoms as spheres and bonds as cylindrical connections between these spheres. This model provides a simple and intuitive visualization of a molecule's atomic structure. It highlights individual atoms and their bonds, including their bond types. However, it may not accurately represent the spatial relationships between atoms in larger molecules or macromolecular complexes.

### 2.1.2 Surface

Surface representations depict the three-dimensional shape of a molecule by displaying its solvent-accessible surface. [citation] This model provides a more accurate representation of the molecule's overall shape and size, making it especially useful for studying macromolecular interactions and the binding of small molecules. Surface models can be displayed as solid surfaces or wireframe meshes.

### 2.1.3 Cartoon

Cartoon representations simplify the molecular structure by focusing on the secondary structure elements of proteins and nucleic acids, such as alpha helices, beta sheets, and loops. Alpha helices are often depicted as a spiral-like structures, whereas beta sheets as arrows.

This type of visualization is particularly useful for visualizing large macromolecular complexes, as it highlights the overall organization and topology of the molecule without the clutter of atomic details. The simplification of the structure also makes it easier to understand the folding and dynamics of the molecule. [citation]

### 2.1.4 Coloring of molecular visualizations

Coloring is an essential aspect of molecular visualization, as it can provide additional information and help to emphasize specific features or properties of the molecule. Some common coloring schemes include: - By element: Atoms are colored according to their chemical element (e.g., carbon in grey, oxygen in red, nitrogen in blue). - By partial atomic charge: Atoms or residues are colored according to their charge or charge sum. Negative charges are depicted in red, positive charges in blue.

## 2.2 Visualization software

There are numerous software tools available for visualizing molecular data, with varying levels of complexity, customization, and features. Two widely used tools are LiteMol and Mol*.

### 2.2.1 Litemol

LiteMol is an open-source, web-native molecular visualization tool that supports various file formats and offers a user-friendly interface for creating visualizations. LiteMol provides essential visualization types, including ball and stick, surface, and cartoon representations, as well as options for customizing colors, lighting, and other display settings. The web-based nature of LiteMol makes it easily accessible and platform-independent.

### 2.2.2 Molstar

Mol* (Molstar) is another web-native molecular visualization tool, developed as part of the wwPDB OneDep system for macromolecular structure deposition and validation. Mol* offers a wide range of visualization options, including advanced features such as electron density maps and validation reports. Mol* supports many file formats, including PDB, mmCIF, and PDBx/mmJSON. Like LiteMol, Mol* is platform-independent and can be accessed from any web browser.

Mol* emphasizes interactivity and offers various tools for manipulating and analyzing the molecular structure, such as distance and angle measurements, selection and display of specific residues, and custom coloring schemes. Additionally, Mol* provides integration with external databases and services, such as UniProt, PDBe, and RCSB PDB, enabling users to quickly access related information and resources.

TODO: add screenshots of Molstar in action + add David citations

# 3 Molstar partial charges extension

Visualizing partial atomic charges in molecules is an essential aspect of computational chemistry research, aiding in analyzing complex molecular structures. Molstar, a popular software tool for visualizing molecules, provides an extensive range of features for users to explore molecular structures. However, the tool lacks the functionality to color and label atoms and residues based on their partial atomic charges. This can be a significant limitation for researchers. In response to this need, we have created an extension to Molstar that addresses this limitation.

This chapter describes the requirements for the extension, the custom mmCIF categories necessary for storing the partial atomic charges, and the implementation of the extension itself.

## 3.1 Requirements

This section briefly describes the formal requirements for the Molstar extension. Firstly, the extension should enable the coloring of atoms and residues based on their partial atomic charges. Secondly, it should describe the charge values of the atoms and residues. Lastly, the extension should be seamlessly integrated into the Molstar library, facilitating access to its features regardless of the Molstar viewer instance.

## 3.2 Custom mmCIF categories

TODO: rewrite

However, in order to store partial atomic charges within a single file, we needed to create custom categories within the mmCIF format. The design of these custom categories was a crucial step in creating the extension since we needed a way to package the charge data together with the structure data into one file.

Without having everything in one file we would have to provide the charges in a different way e.g. through custom import controls.

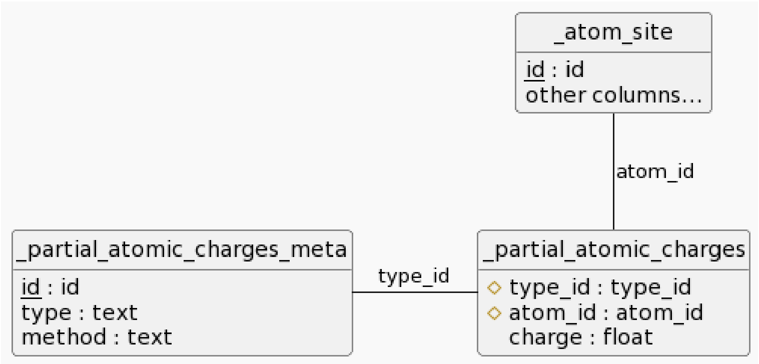

**Figure 3.1:** Diagram of custom mmCIF categories

For this purpose we have chosen the mmCIF file format. There are a couple of reasons already discussed in section 1.2 as to why mmCIF is more advance file format than the other formats discussed in section 1.2. For our purposes the biggest advantage was the ease of creating custom data categories for describing data relating to the partial atomic charges.

The custom categories allowed us to store information about the partial atomic charges separately from the other structural data.

We used two separate categories for storing the charge data. The first category stores the partial charge values for each atom in the structure. The binding of atom and its charge is done using an attribute atom_id which is a pointer to the atom_site.id. item. Additionally, we use an attribute typeId as a pointer into the second category which is dedicated to storing metadata about the charge sets.

The metadata category has three attributes - id, type, and method. Attribute id is a unique identifier for the charge set, attribute type serves to describe the type of computation method used (e.g. empirical, quantum etc.), and lastly the method attribute describes the method name with parameters if applicable (e.g. 'EEM/Racek 2016 (ccd2016_npa)').

Figure 3.1 provides a detailed illustration of the structure of the custom mmCIF categories.

## 3.3 Implementation

The extension consists of multiple providers that are registered in a global registry. Each provider serves a distinct purpose, which will be described in the following subsections.

### 3.3.1 Charges provider

In order to use the charges from the mmCIF file, it is necessary to store them into a model property. Later in the color theme and label providers it will be used to access the mmCIF file data.

The provider works by creating a structure for holding

TODO: add code of the model property TODO: add code of the residue charge sum

### 3.3.2 Partial charges color theme

This provider serves as the central component of the extension, with its primary function being to assign colors to atoms based on their charges. It achieves this by supplying a factory function that generates a ColorTheme object. Within this object, there is a 'color' property featuring a function that assigns a color to a specified location, which could be a structural element, bond location, and so on.

To establish the color for an atom, two color interpolations are employed: one for negative charges and another for positive charges. Atoms with positive charges receive a color from a white-to-blue gradient, while those with negative charges are assigned a color from a white-to-red gradient. See Figure **??** for further details.

### 3.3.3 Labels

This provider creates labels for a specified Loci object. A Loci object is akin to a Location object but is utilized for general selections. Consequently, it is essential to first extract the location from the Loci object in order to obtain the atom ID. The charge is acquired from the property provider, and the label is an HTML string that conveys the charge of the atom or residue.

## 3.4 Molstar integration

## 3.5 Future work

The extension can be improved in many areas - mainly in the UI. It can be optimized by storing the atom and residue charge maps statically. The extension unnecessarily recalculates the maps when switching to another charge set. Another area that could use some work is the UI. When a user wants to switch to another charge set, they need to use the State Tree UI. If the user has only one structure loaded, this is fine. However, once you have more than one structure and want to change the charge set for each one of them, then it becomes very tedious. A better approach is to create a custom UI element in the right sidebar which would serve as a shortcut to set a charge set globally or for each structure individually. Another custom UI control could be used for setting a global max absolute charge - this is also done manually for each structure.

# 4 Molstar viewer plugin

TODO: maybe make this a section in chapter about ACC2

In addition to creating the partial atomic charges extension for Molstar, it was necessary to create a custom Molstar viewer instance to facilitate custom functionality not present in the official Molstar viewer instance deployed at molstar.org/viewer. This chapter focuses on describing the implementation and functionality of the custom Molstar viewer instance for the web applications discussed in chapters 5 and 6.

## 4.1 Requirements

The viewer needs extended functionality

# 5 Atomic Charge Calculator II

## 5.1 ChargeFW2 extension

To accommodate the specified output file format for Molstar, ChargeFW2 required an extension that would output a single mmCIF file containing both the molecular structure and charges. The charges were appended to the end of the file.

## 5.2 Multicharge support

### 5.2.1 Backend

### 5.2.2 Frontend

## 5.3 Molstar viewer integration

# 6 AlphaCharges

## 6.1 Viewer extension

## 6.2 Molstar viewer integration

# Conclusion

Things were created and some things even work.