DOI: 10.1111/cgf.14807
Eurographics Conference on Visualization (EuroVis) 2023
D. Archambault, R. Bujack, and T. Schreck
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 42 (2023), Number 3

ChemoGraph: Interactive Visual Exploration of the Chemical Space

Bharat Kale[1], Austin Clyde[2,4], Maoyuan Sun[1], Arvind Ramanathan[4], Rick Stevens[2,4], and Michael E. Papka[3,4]

[1]Department of Computer Science, Northern Illinois University, USA
[2]Department of Computer Science, University of Chicago, USA
[3]Department of Computer Science, University of Illinois Chicago, USA
[4]Argonne National Laboratory, USA

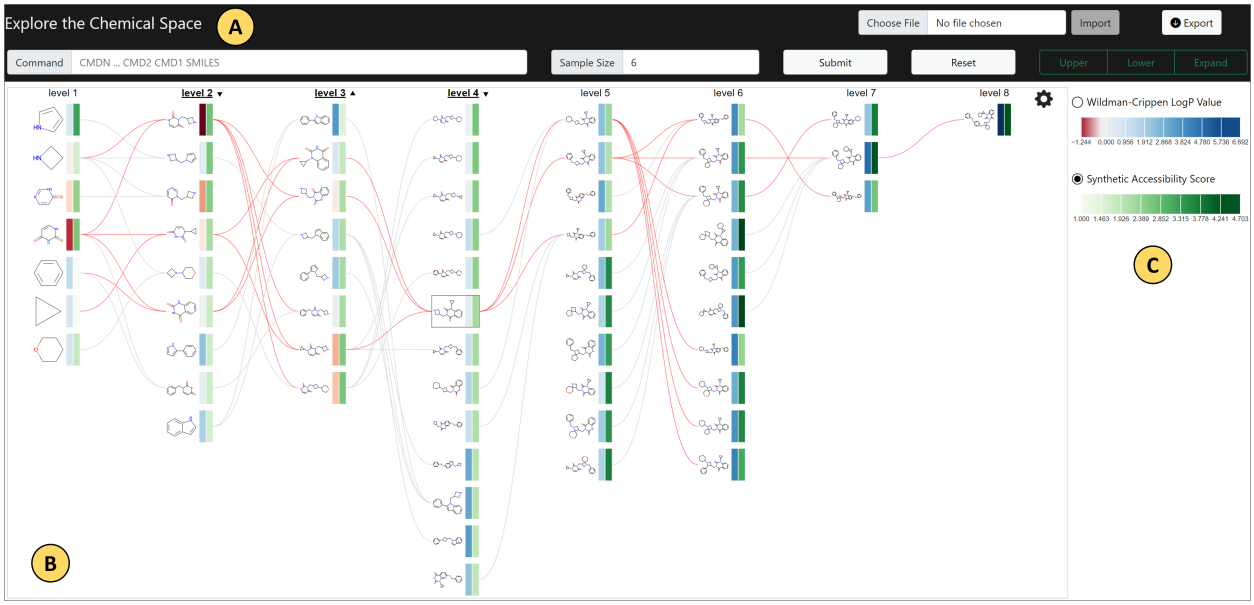


Figure 1: *An overview of ChemoGraph. (A) Control Panel: Provides a way to query for scaffold classes and controls to run graph operations for exploring the space. (B) Visualization: Chemical space organized into scaffold classes that form various levels in hierarchy. Each level, with corresponding scaffold classes, is represented as a list. The edges represent structure inclusiveness among the scaffold classes. Each node is encoded with attributes of interest using color intensity. (C) An interactive legend showing the color scales used for numerical attributes.*

Abstract

Exploratory analysis of the chemical space is an important task in the field of cheminformatics. For example, in drug discovery research, chemists investigate sets of thousands of chemical compounds in order to identify novel yet structurally similar synthetic compounds to replace natural products. Manually exploring the chemical space inhabited by all possible molecules and chemical compounds is impractical, and therefore presents a challenge. To fill this gap, we present ChemoGraph, a novel visual analytics technique for interactively exploring related chemicals. In ChemoGraph, we formalize a chemical space as a hypergraph and apply novel machine learning models to compute related chemical compounds. It uses a database to find related compounds from a known space and a machine learning model to generate new ones, which helps enlarge the known space. Moreover, ChemoGraph highlights interactive features that support users in viewing, comparing, and organizing computationally identified related chemicals. With a drug discovery usage scenario and initial expert feedback from a case study, we demonstrate the usefulness of ChemoGraph.

Keywords: chemical space exploration, cheminformatics, multipartite graphs, data visualization

CCS Concepts

• **Applied computing** → **Chemistry;**

1. Introduction

Chemical space, the property space in cheminformatics including every possible chemical compound, is estimated to be at 10^{60} compounds [BMG96]. For applications such as drug design and discovery and material design, chemical space is a huge domain to explore. This motivated many researchers to create efficient and automated solutions to explore this space with the main goal of identifying novel compounds. Especially in drug discovery, this process is called *scaffold hopping* [SNGS99, HSB17], where we typically start with known active compounds, modify their central core structure, and end with novel compounds. This is an important topic in medicinal chemistry and has various practical applications (e.g., replacing a complex natural product with a synthetic compound that has the same desired activity). *Scaffold hopping* requires analysts to understand the structure of compounds and patterns among how these compounds are connected so that for a given compound, identifying structurally similar compounds in the close neighborhood becomes easier.

Current approaches rely on searching and comparing scaffolds from available chemical databases (e.g., GDB-17 [RVDBR12] and PubChem [KTB*16]). This is highly time consuming given the vastness of the domain and the slowness of the techniques, such as docking, to be computed on the target set. Also, these databases are created as repositories of chemical compounds and are not intuitive for exploring the chemical space as a structured domain. With the advancements in hardware and computing methods, many studies have focused on exploring machine learning- and high-performance-computing-based solutions to scaffold hopping [STS09, CKS*21]. Existing works (e.g., scaffold embeddings [CKS*21]) use transformer models to retrieve molecules, analogous to querying a database, by internally performing scaffold hopping; meaning that instead of storing associated structures of billions of molecules, it computes it on the fly. This architecture lends a natural structure to chemical space, and gives us the ability to use their models in the background to navigate this space like a graph.

As efforts to search chemical space using automation and algorithms has increased, so has interest in applying visualization techniques to the task [NMF19, ORO*15, NMF17]. Starting with a set of known compounds of interest, exploration of related compounds quickly becomes a complex web of relationships. Consider the following drug discovery pipeline scenario. Sarah, a chemist, finds a lead compound *A* for a new drug. Her next step is to identify a series of pre-clinical candidates. She is using a machine learning model [CKS*21] that can generate a set of compounds that are related to *A*. After several iterations, the resulting set contains compounds at various levels from *A*, whose relationships are becoming increasingly difficult to track. Sarah needs support to make sense of the identified space and to make decisions on which compounds to use based on various properties for growing the space further. To address this challenge and support the analysis tasks of chemists during drug discovery, we created ChemoGraph, a novel visual analytics technique to interactively explore chemical space. Having a visual representation of the navigated space helps reduce the cognitive load on the analyst and enhances their comprehension of large amounts of data.

In summary, our key contributions in this paper include:

1. We formalized the drug-like chemical space by representing it as a hypergraph.
2. We combined an interactive interface with machine learning models to help users dynamically grow a chemical database. This dynamic expansion allows for exploration beyond the boundaries of a specific dataset.
3. We designed interactive techniques to explore chemical space as a structured domain, starting from a set of known compounds.
4. We conducted a case study with domain experts to understand the usability of ChemoGraph.

ChemoGraph is different from previous works in that there are no set bounds to the dataset used in the exploration. A user can start with a subset of chemical space and if at any point the system runs out of samples from that subset, ChemoGraph's computational backend will compute novel compound classes and their relationships to enrich the coverage of the space. Also, the chemical space visualization problem can be generalized to the problem of finding and analyzing relationships among sets of different entities. Analyzing the relationships among scaffold classes from various levels is similar to analyzing the relationships among entities in different domains such as tables in a relational database and gene sharing in the microbial world [CLMB16].

2. Related Work

The underlying structure of chemical space is a graph where nodes can be grouped into disjoint sets, and edges connect to nodes from adjacent sets when sets are ordered according to the inherent hierarchy. Thus, this problem can be considered a graph exploration problem. So we reviewed visualization designs for graph exploration and related work in exploring and visualizing chemical space.

2.1. Graph Exploration

Graphs are powerful data structures with a strong presence in many application domains [SMS*17] and generally consist of nodes and links. Many real-world graphs also need the ability to present multiple attributes simultaneously to reveal the relationships among them. Such graphs are called multivariate graphs [NMSL19]. Other relevant variants of graphs include multipartite graphs and hypergraphs. A multipartite graph is a graph whose vertices can be partitioned into disjoint sets [GJ79]. A hypergraph is a graph in which an edge can connect any number of nodes [FFKS21]. Our work deals with a combination of the three variants of graph structures. Typically, we choose a layout to visualize graphs based on various characteristics, but mainly topology. The most intuitive and commonly used layouts for visualizing graphs are *node-link diagrams*, *matrix diagrams*, and *list views*.

Node-link diagrams are useful for navigating large heterogeneous and multivariate network data. PivotPaths [DRRD12] uses facets to organize data into individual sets. VIGOR's fusion graph [PHE*17] uses node-link diagrams to help understand patterns in a subgraph resulting from querying a graph database. Unlike PivotPaths [DRRD12], where the partitions based on entity types are explicitly revealed spatially, VIGOR [PHE*17] only uses color and suffers more from visual clutter. Node-link diagrams are also used to visualize hypergraphs by introducing a new type of node to deal

with the encoding of a hyperedge that can connect more than two nodes [FFKS21]. All these studies support a rich set of interactions as visual clutter increases exponentially with the increase in data in node-link diagrams. Interactions to filter and highlight interesting regions play a key role in sensemaking of the data in such scenarios.

Matrix diagrams are preferable to present bi-clusters [HSBW11, KKC*04, BTBC*21, SNR14, Sun16] by rearranging columns and rows. In multipartite graphs, relationships are confined to entities between different domains. Because of this characteristic, matrix representation of multipartite graphs create sparse matrices. Prior works (e.g., Bixplorer [FSB*13,SBNR14], Furby [SGG*14], MissBiN [ZSCC20,ZSCC19], and NodeTrix [HFM07]) use matrices to reveal local subsets while using other layouts for global structure.

List views use individual axes to show nodes in each set [SGL08, PLS*14,SMNR15,SJUS08]. Jigsaw's list view [SGL08] allows for exploration of relationships between lists of entities via visual links and color. ConTour's list view [PLS*14] combines faceted search and interactions to highlight related entities using position and color of an item in a list. The main difference between Jigsaw's list view [SGL08] and ConTour's list view [PLS*14] is in how the relationships between entities using explicit links are shown. Schulz et al. [SJUS08] also employ a list view for visualizing bipartite networks in biology. They use tables to represent a list so that each list item can have multiple cells to encode multiple attributes and the relationship between tables is explored using visual links. Combining positives from both Jigsaw [SGL08] and ConTour [PLS*14], BiSet [SMNR15] and MERCER [WSM*18] uses explicit links and interactions to align related entities [SZW*18]. PolyViz [UM18] takes a slightly different approach by using a radial list view. List views are also popular with hypergraphs, especially to represent the dynamics in a network's structure [VBP*19].

ChemoGraph uses list view as the primary layout with explicit links. More discussion about this design rationale is in Section 4.3.

2.2. Chemical Space Visualization

Visualization and analysis of chemical space has multiple applications in drug discovery (e.g., lead optimization, virtual screening, and comparing compound libraries [MFMMG*08]). As a result, numerous studies have explored the visualization techniques and applied them to the domain of computational chemistry for representing and navigating chemical space [NMF17, NMF19, SK21, SFEJ15,HŠVS14,SER*07]. Osolodkin et al. [ORO*15] and Wawer et al. [WLWB10] conducted comprehensive reviews on the types and applications of visualization techniques for data analysis tasks in chemical space. We can categorize these approaches into two key groups based on techniques used: *descriptor vectors* and *networks*.

The primary approach behind visual representations that use *descriptor vectors* is dimensionality reduction, using techniques such as principal components analysis (PCA), t-distributed stochastic neighbor embedding (*t*-SNE), and self-organizing maps. Due to the inherent high-dimensional nature of chemical space, such representations are mapped to a two-dimensional space and are plotted using scatter plots. ChemMaps [NMF17] uses PCA for visualizing correlation between compound datasets. FragNet [SK21] computes molecular similarity among huge databases and visualizes the distribution of molecules by applying *t*-SNE. Naveja et al. [NMF19] introduced constellation plots by identifying groups of compounds using *t*-SNE for interpreting structure-activity relationships in chemical space. A common problem with this group of approaches is that they lack interactivity. Few works that support interactive analysis [SUS*20] only allow for exploration of screened compounds. While the objective of ChemoGraph is to allow users to navigate the unknown space around a known compound by dynamically growing the local neighborhood.

Approaches that use *networks* leverage interactive visualizations that support exploratory workflows. Kakar et al. [KQR*19] claim that networks tend to show association relationships better. They created a visual analytics system, DIVA, to analyze candidate drug interaction signals via coordinated views of force-directed graphs and tree views. Sushko et al. [SNK*14] created a transformation graph to reveal the web of transformations on a compound that affect specific properties. DataWarrior [SFvKR15] creates visual representations by using node similarity for node placement. Another variant in network-based visualizations use directed edges to show a hierarchy in the data. Scaffold tree [SER*07] introduces hierarchical classification of scaffolds or classes of compounds to visualize tree structure that depicts the parent-child relationship between scaffolds and compounds. The main problem in these studies is that they lack the ability to enumerate the chemical space. To our knowledge, none of the prior works leverage structure and relationships between compounds. Hence, we are lack of effective tools to help understand and navigate the chemical space. Molpher [HŠVS14] tries to achieve this to some extent by exploring possible paths that connect a source compound and a target compound. This exploration limits the space only to paths between the source and the target. Our work differs from all the prior studies in that we are not limited to a specific compound library. Instead, we use a compound library for preliminary search results and extend it using transformer models, theoretically to search the entire drug-like chemical space. However, we only claim that our technique allows searching the entire chemical space to identify candidate compounds, not to visualize it. This is because chemical space is enormous and showing the full overview is expensive. Also, medicinal chemists are mainly interested in exploring local neighborhoods of compounds of interest rather than obtaining an overview of the entire space.

3. Background and Domain Goals

Small molecules act on proteins in cells. Through the introduction of a small molecule into a cell, the behavior of proteins can be modulated. In this way, biological response (BR) can be considered as a function of chemical structure (C), $BR = f(C)$ [Han76]. The magnitude and specificity of this action are determined by attributes of the compound such as its shape and flexibility, polarity, and physical properties (e.g., solubility) [Rey15]. Shape and flexibility refer to the range of three-dimensional *conformations* that a compound can take on, where flexibility refers to the likelihood of multiple stable states or an overall instability. Polarity refers to the distribution of charges of chemical groups composing a small molecule, leading to a molecule with an *electric dipole moment*—an uneven distribution of charges. The idea of *quantitative structure-activity*

relationship is to characterize how changes in these parameters lead to changes in biological response. For example, given two compounds C_1 and C_2, we might decompose the change activity around these attributes of the structure, $\Delta BR = f(C_2) - f(C_1)$.

$$\Delta BR \approx f(\Delta(\text{sterics}) + \Delta(\text{polarity}) + \Delta(\text{hydrophoic}) + C_1) \quad (1)$$

This is a core idea of medicinal chemistry; that small changes in chemical structure should produce discernible and small changes in response. Current computational workflows which use generative models for virtual screening of drugs do not take into account this relationship between compounds. Thus a driving goal for computational chemistry workflows is organizing that relationship.

Chemical space. Many problems ranging from drug discovery to material design require identifying molecules from a design space of all possible chemical compounds. This space is called *chemical space* [Rey15]. Estimates place the number of compounds in chemical space at around 10^{60} [BMG96]. Thus, it is infeasible to explore all the compounds in this theoretical space. While the overall design space is intractably large, medicinal chemists often work with targeted libraries consisting of compounds that may be easily synthesized. One benefit to targeted libraries is that they are small enough to be tractable for virtual or experimental high-throughput screening. Furthermore, targeted libraries are often closely related in chemical space with a similar chemical moiety. However, over time, reliance on the same chemical datasets has resulted in bias in drug discovery and other fields due to the lack of diversity in the compounds screened [JLH*19]. Recent evidence has also strongly indicated that large, diverse screening libraries have an advantage over less diverse and smaller libraries [LWB*19].

In order to overcome this bias, two main approaches have been explored. Recent work in generative drug design has attempted to get around the bottleneck of only enumerated compounds by generating likely novel compounds on the fly [GMH*18]. Some of these techniques are guided by surrogate models in an inverse-design setup [SLAG18]. However, even with generative models, a problem emerges. If one can generate billions of compounds, how can this be organized? The second approach, which may use compounds from a generative model, includes increasing screening library sizes, thus moving away from targeted libraries to vast subsets of chemical space. These libraries can be enumerated computationally and can reach into the tens of billions [dSROFS19]. As library sizes increase, the tractability of computational screening decreases. However, the scaling of chemical libraries is assumed such that every molecule is an independent sample. This assumption, which is used in cheminformatics, seems implicit and there is evidence that relationships between compounds can be exploited for virtual screening tasks [WQT*21].

Chemical Scaffolds. Chemists identify the shape of compounds through common substructures. The Bemis-Murcko decomposition of a molecule breaks a molecule into four parts: ring systems, linkers, side chains, and the scaffold. *Chemical scaffolds* are defined then to be the union of ring systems and linkers effectively capturing the common core of a compound without the side chains. Figure 2 illustrates this decomposition. Scaffolds are well defined computationally and offer a general description of global properties (e.g., orientation in a protein binding region) [BM96, SC20].

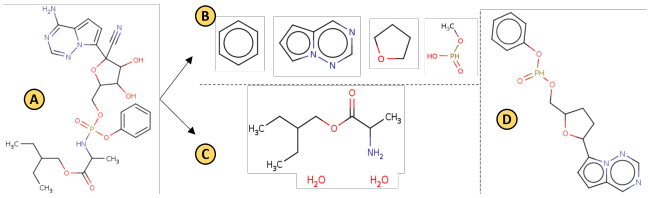


Figure 2: *Decomposition of a sample molecule Remdesivir into components. (A) chemical structure of Remdesivir. (B) rings and linkers. (C) side chains. (D) The rings and linkers are combined to create a scaffold.*

By grouping compounds that share the same chemical scaffold, a hypergraph is created. Two compounds are said to share a scaffold-relationship if the corresponding scaffolds are the same. On top of this, scaffolds can form a graph themselves. Scaffolds can be decomposed by breaking up linkers or fused rings. We call this relationship *lower* as the decomposed scaffolds have a lower number of rings. The flip side of the relationship is called upper, as the upper set of scaffolds for a given scaffold is all scaffolds that contain it, and thus have more rings. In Figure 3, we illustrate the hypergraph relationship between molecules and their scaffold with a circle. We depict the relationship between *upper* and *lower* through the areas.

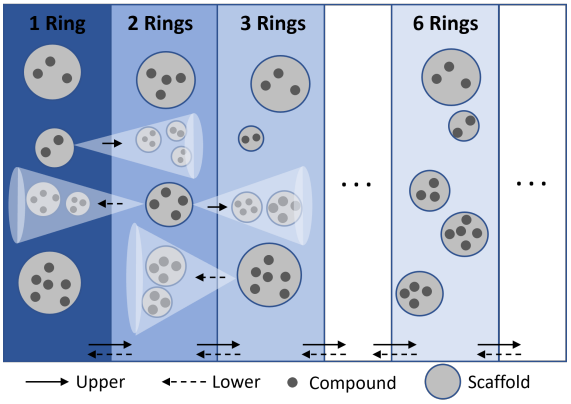


Figure 3: *Scaffolds as classes of chemical compounds in the chemical space. Every chemical compound belongs to a single scaffold class. Scaffold classes are connected by common substructure thus forming a hierarchy. Upper and Lower are operations to traverse the scaffold classes at different levels of hierarchy.*

There are two main domain goals that ChemoGraph addresses. The first is to create a platform for scaffold-based drug design [BFS04]. Such a platform encourages medicinal chemists to think through chemical space through scaffolds. By examining and incorporating different data such as properties or experimental screening results, medicinal chemists can interact with local regions of chemical space while at the same time using generative models to grow and increase the diversity of their molecular series. The second goal aims to address the problem of the data deluge. Given the billions of compounds medicinal chemists want to explore, a platform which can both be interactive and run state-of-the-art generative drug design models attempts to bridge a purely computational approach to a more traditional rational drug design program.

4. Design Requirement Analysis

This work is the result of a collaboration with two computational chemists from the Argonne National Laboratory and a doctoral student from the University of Chicago. The overarching goal of our collaborators is to allow interactive enumeration of chemical space and to better understand the relationship between compounds within that space. Our collaborators currently use a command line utility to run various operations for enumerating the space. However, they recognize that using this approach to make sense of the enumerated space is challenging. This created a need for a visualization tool that supports the analytical tasks that generate novel compounds on demand.

The initial design phase had a series of collaborator meetings to understand the problem and to gather insights on what each individual would like to *see* and *perform*. These discussions helped to define two key artifacts: 1) hypergraph abstraction of the chemical space, and 2) operations used to expand the space. Next, the team focused on key aspects of the workflow, which resulted in a set of abstract tasks (Section 4.2). Based on these tasks, we explored layout options for representing and navigating the space (Section 4.3).

4.1. Chemical Space as a Hypergraph

Hypergraphs provide a natural way to model complex group relations in data. In chemical space, compounds are grouped into scaffolds and an edge connecting two scaffolds represent relationships between multiple compounds in the scaffolds (i.e., a hyperedge). Thus, we formalize the chemical space as a hypergraph and nodes (scaffolds) are defined at various levels as illustrated in Figure 3. This depiction enables us to effectively capture the relationships that exist only between the consecutive levels while effectively using the space to present the complex encoding of the entities. Simple chemical scaffolds with few rings (e.g., 1 and 2 rings) are represented in the hierarchy's lower levels. While complex scaffolds with a larger number of rings are displayed in the higher levels. The operations that identify related scaffold classes from lower levels or upper levels, for a given scaffold class (illustrated using cones in figure 3), are used to grow and navigate the space.

4.2. Tasks Driven by Requirements

After iterative discussions with collaborators and domain experts over the past year, we identified a set of tasks analysts need to perform in order to achieve the previously described domain goals. An important note here is that the entire process is highly exploratory. Thus, to achieve any of the domain goals, a combination of the following tasks must be iteratively executed.

- **Identify related entities (T1):** Given a node A, (a) identify all directly or indirectly related nodes of A, (b) compare related nodes by attributes, and (c) understand the interaction states of node A. An example of this task is to identify all compounds that are in the lower or upper cone of given compound A.
- **Rank entities (T2):** In multivariate scenarios, personalizing the arrangement of entities in the network based on some attributes is an important task to choose entities of interest. An example is to pick a compound that can be easily synthesized by rearranging entities in ascending order of SA score.
- **Explore entities across levels (T3):** It is critical to allow users to interactively control the direction of growth of the space. This can be achieved from the following sub-tasks: users choose an entity of interest A based on presented information to create (a) superstructures, (b) substructures from A, and (c) identify paths between entities that are separated by multiple levels to quickly make sense of the relationships in the network.
- **Explore individual entity class (T4):** Given an entity A, identify the samples that class A covers. An example task is given a scaffold A, find a set of chemical compounds that belong to the scaffold class A.
- **Identify structure similarities (T5):** Given an entity A, identify similar but complex entities. For example, given a scaffold A, highlight scaffolds in higher levels with A as the substructure.
- **View more samples (T6):** Given an entity A, identify more samples of A's related entities. The background models that run during the operations like Upper use sampling to control the result size. A user is interested in looking at more samples without running the operation again. An important task is to look at more samples in the neighborhood of the focus area without running the computational model frequently.

4.3. Layout Selection

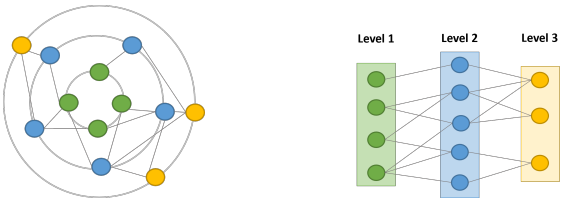
Figure 3 shows a conceptual abstraction of chemical space, wherein chemical compounds are organized by levels. Such conceptual levels are based on the structure (i.e., number of rings) in a compound. Considering this, matrix-based layouts may not work well, as they can result in highly sparse matrices.

Moreover, matrix diagrams are not intuitive for organizing more than two domains. Thus, we considered two approaches to organize the chemical space: node-link diagrams and list views.

They both provide control over layout, enable a way of separating nodes and edges visually, and are widely used to support visual analysis and present information based on types [SGL08, SMNR15, ZSYN15, ZSCC17]. However, compared to node-link diagrams, list views are more effective for context slicing of visual entities based on conceptual abstraction (e.g., different types of information) and for spatially organizing them. This approach offers better navigation of the space, a main design consideration of ChemoGraph. Specifically, list views highlight an advantage of supporting systematic information searching both vertically and horizontally, which corresponds to explorations within the same structure-level and across different levels. Hence, we chose list view as our base layout. Within the list view, we explored both radial arrangement and linear arrangement of nodes as shown in Figure 4.

Radial layout allows arranging items from different domains along concentric circles such that all items from one domain are placed along one circle. This layout utilizes the space effectively but does not work well when showing relationships. It creates more edge crossings even with a relatively small data set.

Linear layout organizes items from different domains along parallel axes (e.g., parallel coordinates [ID09]) such that all items from one domain are placed along one axis as a list. It requires vertical scrolling when the number of nodes increase, but it clearly manages the separation of entities from different domains even when edges



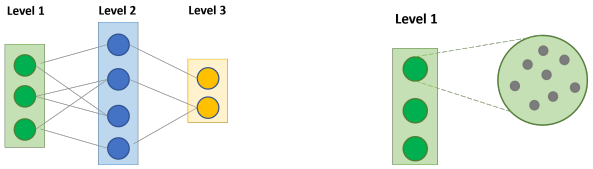
(a) *Radial view* **(b)** *Linear view*

Figure 4: *Alternative list-based layout designs.*

are shown. As edge crossings occur only between consecutive domains, we can use techniques such as edge bundling [SMNR15] and ordering [SZW*18, KSP21] to reduce visual clutter.

4.4. Level-Based Exploration

There are two types of relationships that need to be supported (Figure 5). The first type focuses on cross-level relationships (i.e., the direct or indirect relationships between entities across levels). They indicate the global structure of the covered space. The second one focuses on within-entity relationships i.e, the relationships between an entity and the items that belong to it. They indicate individual scaffold classes and their compounds.



(a) *Between entities* **(b)** *Within an entity*

Figure 5: *Types of relationships.*

5. ChemoGraph

The tasks discussed before (Section 4.2) are driven by a highly exploratory analysis process that requires a system that can flexibly handle both existing chemical databases and models that generate new information.

5.1. System Overview

Figure 6 shows an overview of our approach. The *database* component is constructed from the pre-computed graph of scaffold classes, chemical compounds, and their associated relationships. There is a computational component in the backend comprised of *transformer models and algorithms*. Given a scaffold, sub-scaffolds can be found with a graph algorithm [CKS*21]. However, super-scaffolds often require a database search or sampling of chemical space. In order to grow the number of accessible super-scaffolds, a transformer model was used which can generate larger scaffolds that contain the input scaffold [CKS*21]. A different transformer model is used to enumerate compounds within a scaffold class as well [CKS*21]. This is a heterogeneous backend as sometimes a query requires a graph algorithm computation, a database search, or a query to a generative model. The contents of the database along

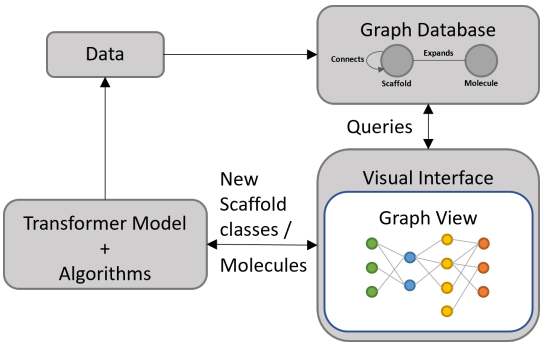


Figure 6: *The system overview of ChemoGraph. Data created using transformer models and other computational algorithms in the form of a graph is ingested into a database. A visual interface communicates with both the database and the models to show the graph.*

with the results from the computational component are shown to users in the *visual interface* with an interactive graph visualization.

5.1.1. Computational Backend and Graph Operations

The different operations (lower, upper, scaffold, and expand) entail different types of computation. An example showing the results of running each of the available graph operations on a given input is shown in Figure 7.

Lower and scaffold are deterministic graph operations. Given a molecule, its scaffold can always be computed directly from the structure using cheminformatics packages (e.g., RDKit [Lan13]). Similarly, given a scaffold, its constituent components, lower scaffolds, can always be directly computed from the scaffold by breaking linkages between rings. These operations are fast as they are simple graph algorithms. They are also deterministic in the sense that they will always produce the same set of results without any hyperparameters. A chemical compound is the only input required for performing these operations.

Upper and expand are generative operations. They are like inverse operations to lower and scaffold operations, respectively. For example, given a scaffold, the expansion of the scaffold samples the set of molecules with the given scaffold. Unlike the scaffold operation, this is a generative task because the set of all molecules with a given scaffold is not known at runtime and would be intractable to compute fully. Likewise, given a scaffold, the set of scaffolds that contain the given scaffold as a substructure constitutes the upper set. A simple way to see how this cannot be directly computed is by imagining the set of upper scaffolds of a simple benzene ring. In a sense, the upper set of this six member ring would be a large majority of the chemical space. It would be computationally intractable to enumerate such a set. Thus, upper and expand operations sample the underlying large sets in order to provide an exploratory mechanism. Hence, the inputs required for performing these operations is an input compound and an expected sample size. The details of the models used and how they are trained are available in [CKS*21].

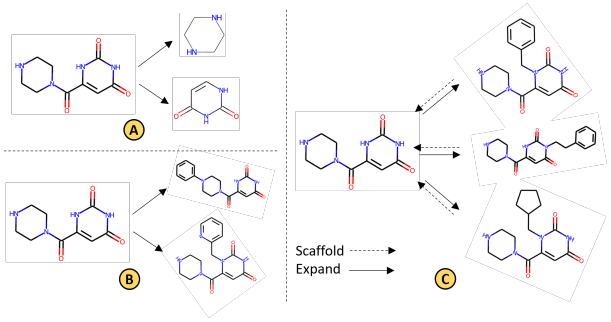


Figure 7: *An example of lower, upper, expand, and scaffold operations: (A) A scaffold is decomposed into sub-scaffolds using the lower operation. (B) Super-scaffolds are identified for a given scaffold using the upper operation. (C) Retrieving chemical compounds from a given scaffold using the expand operation and, vice versa, the (scaffold) operation.*

5.2. Visual Interface

The goal of the visualization is to not only present chemical space but also perform scaffold hopping (i.e., to start from a set of chemical compounds of interest and allow the user to interactively grow the space by choosing new compounds of interest from the visible space based on various properties). There are four main aspects to ChemoGraph's visual interface: *specifying input to start the exploration*, *visual representation of compounds using scaffold classes and graph operations*, *interactions to support exploration*, and *export to save the progress*. In this section, we describe them in detail and explain the design decisions behind the visual representation.

5.2.1. Specifying Input

ChemoGraph's visual interface allows specifying inputs in two different ways: Using *Commands*, users can provide expressions supported by our command line utility. These expressions consist of combination of graph operations followed by a chemical compound or a scaffold class as a SMILE string [Wei88]. Users can use commands to start their exploration from a single compound/scaffold class of interest. In addition, *files* can be used to set up the initial view by importing either a previously saved view or a set of compounds/scaffold classes. ChemoGraph supports JSON and SMI file formats as input. Once the initial view is specified using the above methods, users can either continue giving more command expressions to update the graph or interactively use the controls representing various graph operations on a selected node.

5.2.2. Visual Representation

Nodes in the graph represent scaffold classes and edges represent relationships between these classes. The visual representation of a node is chosen based on the information that needs to be encoded on it. Each node has an associated chemical structure and a set of numerical attributes that help in organizing the nodes as well as understanding their characteristics. We use color and position visual channels to encode the information on nodes and edges. Figure 8 shows an example of visual encodings in ChemoGraph. Each node has a Scalable Vector Graphics (SVG) thumbnail to show the chemical structure of the scaffold class and a set of rectangular strips filled with color whose intensity represents the corresponding numerical attribute of the scaffold class (for *T1(b)*) as shown in Figure 8(A). Edges in the ChemoGraph are created using Bézier curves, as they improve the aesthetics of the visualization over straight lines.

In ChemoGraph, color is used to encode numerical attributes on nodes and to differentiate between the possible states of nodes and edges. Depending on the range of attribute values, the color scales used in the visualization are configured between sequential and diverging automatically. Figure 1(C) shows an example of the legend with both diverging and sequential color scales. There are four possible node state options - *normal*, *mouseover*, *selected*, and *related*. A change to node state is distinguishable by a change to the node's border color (for *T1(c)*) as shown in Figure 8. *Normal* state (Figure 8(A) - no border) is used on all nodes by default. When a user interacts with a node, its state changes to *mouseover* (Figure 8(D) - blue border) if the user hovers a mouse on it, or to *selected* (Figure 8(C) - black border) if the user selects it. The state of all the nodes that are directly or indirectly related to the selected node changes to *related* (Figure 8(E) - gray border). For edges, there are two possible states - *normal* and *highlighted*. In Figure 8, (B) and (F) represent *normal* and *highlighted* states of edges, respectively.

The layout of the nodes is critical for organizing scaffold classes within each level. Scaffold classes that belong to a level are placed vertically in a list while levels themselves are arranged horizontally (for *T1(a)*). The ability to arrange scaffold classes based on various numerical attributes help in understanding different characteristics of the classes. We use a legend to select the attribute based on which ordering is performed. Any entry in the legend can be interactively selected as is shown in Figure 1(C). By default, scaffold classes are arranged in the order in which they are created. Once a numerical attribute is selected, a user can order the scaffold classes in each level independently, in ascending or descending order (for *T2*). The label of the level is used to directly manipulate the ordering of scaffold classes in the level by switching between the possible states of ordering (order by creation, ascending order of a selected attribute, descending order of a selected attribute). In Figure 8, (G) shows a level where its scaffold classes are in default order (i.e., the order of creation), (H) and (I) represent levels where corresponding scaffold classes are in ascending and descending order of a selected attribute, respectively.

5.2.3. Interactions for Exploring the Space

ChemoGraph uses graph operations (*lower*, *upper*, and *expand*) to build the graph from the underlying chemical space. The visual interface has controls (Figure 1(A)) to perform these graph operations on selected scaffold classes. *Lower* operation computes the sub-scaffolds (for *T3(b)*) and thus helps in growing the space toward lower levels. *Upper* operation computes the set of super-scaffolds (for *T3(a)*) and helps in growing the space toward higher levels. *Expand* operation computes chemical compounds present in a scaffold class (for *T4*) and helps in exploring the entities in a group. Figure 8(J) shows an example of expand operation where the highlighted substructures in the resulting compounds indicate the structure of the scaffold class they belong to.

ChemoGraph allows users to directly interact with the scaffold classes for making sense of relationships among scaffolds. Users

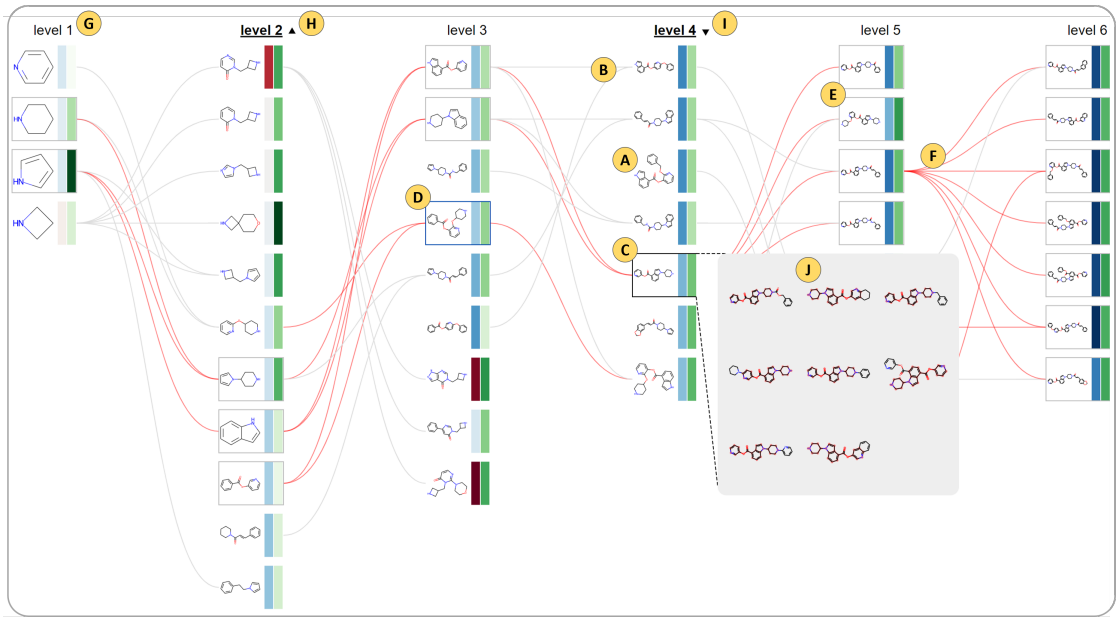


Figure 8: *Visual encodings: (A) and (B) are normal state of scaffold classes and relationships among them. (C) shows the selected state of a scaffold class. (D) is the mouseover state of a scaffold class. (E) shows both directly and indirectly related scaffold classes of the selected class. (F) represents highlighted edges. (G), (H), and (I) show the default order, ascending order by a selected attribute, and descending order by a selected attribute, respectively. (J) reveals the expand operation to show a sample of chemical compounds from a selected scaffold class.*

can interact by hovering over them or by selecting them, to reveal relevant scaffold classes (for *T3(c)*). Hovering only reveals directly related scaffold classes whereas selection allows both directly and indirectly related scaffold classes across all the levels (Figure 8). This means, upon selecting a scaffold class, ChemoGraph allows users to identify both the simple scaffold classes that can be used to synthesize the selected class and complex scaffold classes that can be synthesized from the selected class.

Zoom operation is used to show more details and substructure relationships among the scaffold classes. In ChemoGraph, zoom has flexible settings to *enlarge* and *enrich* the space. *Enlarge* involves scaling. This is useful when looking at the chemical structures of individual compounds/scaffolds. To *enrich* the space, ChemoGraph offers two settings: **adding details** and **adding samples**.

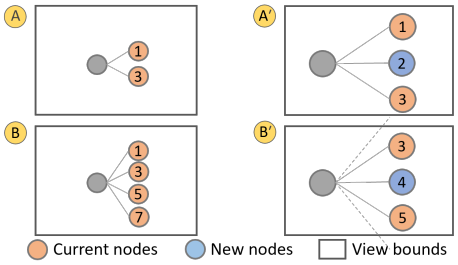


Figure 9: *Illustration of samples on demand: (A) and (A') show a case where both existing and new samples are visible in the view. (B) and (B') present a case where some of the existing nodes go out of the view and new samples are added in between the visible ones.*

Adding details highlights the selected scaffold class as the substructure, if present, in all the scaffold classes from the higher levels (for *T5*). This is the default zoom setting that takes effect when zoom scale is greater than 2. It can be set to turn on/off. **Adding samples** is used to show more samples of scaffold classes in between the currently visible scaffold classes. It is disabled by default and can be enabled at any point. Similar to the paradigm of *"details on demand"*, *adding samples* follows "samples on demand" to allow users to see more samples without running the background models (for *T6*). Whenever the *upper* command is run, the system fetches more samples than what was requested and buffers them. During the zoom-in operation, if there are any buffered samples in the neighborhood of visible scaffold classes, they will be added to the view. Figure 9 illustrates two cases during the zoom operation: zooming-in results in all current nodes still in the view (A) and a few nodes go out of the view (B). Correspondingly, how the system adds new samples is represented by (A') and (B').

5.2.4. Export to Save the Progress

It is necessary to save the SMILE strings of the scaffolds for further usage and the entire view to easily resume the exploration from a given point. ChemoGraph supports both. **Export Smiles** exports the current nodes in the graph to an SMI file where each row represents a SMILE string followed by the encoded chemical properties. **Export View** exports the current view to a JSON file so that experts can import the saved view and resume their exploration.

6. Usage Scenario

We present a usage scenario to demonstrate how ChemoGraph can help an analyst develop a structure-activity relationships (SAR) se-

ries starting from a single lead compound. After high-throughput virtual screening studies have identified a lead compound, the next stage in the drug discovery pipeline is lead optimization. This process involves the enumeration of local chemical spaces, determining a set of compounds based on structure-informed hypotheses, and experimentally testing them. The resulting SAR series allows chemists to determine which groups of the compound are essential to explain the compound's activity, while also attempting to find the most potent compound.

Sarah is a chemist researching antivirals for Coronavirus disease. She starts with an established drug target, MCULE-5948770040, a lead compound determined through high-throughput virtual screening of the SARS-CoV-2 main protease (M^{pro}) [CGK*21]. Sarah aims to develop a SAR series based on this lead compound. Standard practice for generating SAR series entails by-hand enumeration of different possible modifications to the compound locally, maintaining the scaffold within a few hops. For example, Kneller et al. [KLG*21] performed expert SAR on the same lead compound. Using ChemoGraph, Sarah begins her exploration by loading the lead compound using its SMILE string representation. The compound belongs to a scaffold class in level 3. She runs *expand* operation on the scaffold to look at sample compounds in the class. Next, she grows the scaffold space by first decomposing the scaffold (*lower*) and then generating larger scaffolds (*upper*). Now Sarah performs the *upper* and *lower* operations iteratively to produce a SAR series. Without ChemoGraph, Sarah would have to remember explored possibilities and a complex web of relationships to manually generate a single SAR series.

Sarah decides to explore the properties on the scaffolds and organizes the scaffold space by LogP value. The LogP value of a compound refers to its lipophilicity (i.e, ability to dissolve). Sarah uses compounds with high LogP value to grow the graph thus resulting in scaffolds with good lipophilicity scores. Now Sarah wants another SAR series where the compounds in the resulting series can be easily synthesized. She can organize the levels by using the synthetic accessibility (SA) score. The higher the SA score, the more difficult it is to synthesize the compounds. Organizing the scaffolds by ascending order of the SA score allows easy-to-synthesize scaffolds to appear at the top of the list. She starts from the lead compound and creates another series by picking the scaffolds with low SA score for growing the graph. The total range of SA scores for all the scaffolds in the view can be seen from the legend corresponding to the SA score. ChemoGraph allows analysts to control the generation of a SAR series by organizing the scaffolds using different molecular properties and then selecting scaffolds of interest as sources to grow the graph at each step. Sarah can also see that by selecting a specific scaffold, ChemoGraph highlights all the related scaffolds across levels. The ability for the analyst to visually inspect compounds and their relationships while integrating chemical property models is a unique aspect of this workflow. Chemists often use visual information for drug design [FSSL21]. By displaying the molecular properties alongside the compounds, chemists can pursue leads based on optimizing certain molecular properties while tapping into their own expertise to avoid over-optimization or unreasonable compounds, an often-cited critique of solely using generative models without visual inspection [MFB21].

7. Initial Expert Feedback

We conducted a formative case study to assess the effectiveness of ChemoGraph in aiding medicinal chemists in performing various drug discovery tasks. We recruited 8 domain experts, 5 male and 3 female, aged 28-50 (μ=35.86, σ=9.04). To improve the diversity, we invited experts from various research groups. They all are well-familiarized with drug discovery tasks and have backgrounds in computational chemistry.

We conducted the study online asynchronously. To familiarize the participants with ChemoGraph, we sent them a tutorial video showing various features of the interface. After viewing it, participants were asked to explore the interface for 5 minutes before starting on the actual tasks. For the study, we specified 3 tasks that required applied domain expertise on chemical structures and properties for exploring local neighborhoods in chemical space. Task 1 (T1) required participants to construct a small neighborhood of chemical compounds of interest and explore the space using 3 different property prediction models using ChemoGraph. Upon completion, participants evaluated the models and ranked them based on accuracy. Tasks 2 (T2) and 3 (T3) required participants to develop a chemical series of at least ten compounds from a given lead compound and a fragment screen, respectively. Upon completion, participants were asked to return their developed chemical series. Lastly, the participants completed a post-study questionnaire with multiple choice (Table 1) and open-ended questions. The multiple-choice responses were collected using a 5-point Likert scale.

For each task, we computed completion times of each participant and then calculated means and 95% confidence intervals. There are currently no tools to visualize the predictive model landscape of molecular properties and then use them in exploratory tasks to pursue leads. Hence, for task T1, we only looked at how much time participants spent exploring the models using the tool and their responses from the questionnaire. Participants spent an average of 7m 56s. For tasks T2 and T3, the current approach is a by-hand enumeration of the possible modifications to input compounds while following specific rules based on how close the new compounds should be to the input. This task is highly time consuming. However, the completion times that our participants achieved were impressive, averaging a time of 4m for T2 and 3m 54s for T3. This finding was also confirmed by the responses to the questionnaire and in feedback comments (Q5 in Table 1).

Overall, the experts rated ChemoGraph positively. Their responses to the questionnaire (Table 1) clearly suggests that they found ChemoGraph useful for exploratory drug discovery tasks. Specifically, ChemoGraph's ability to use custom models as input and encode them on chemical compounds was highly appreciated (Q4 in Table 1). When asked to specify reasons to use/not use the tool in their own workflows, participants' responses included: *"Excellent deconstruction of a compound. Convolutions help in design and synthesis"*; *"The UX and concept is super smart and easy to master"*; *"Useful for my hypothesis generation and diversity oriented testing in the wet lab"*; and *"I may use the tool because of its ability to extend the compound dataset"*.

The experts also provided suggestions for improving ChemoGraph. One participant noted *"If the tool can also link congeneric*

compounds automatically, it would be very useful." Other suggestions included *"It will provide more flexibility if users can delete the compounds they don't want"* and *"It is really helpful to add metadata of the molecules such as name, database or wikilink."*

Table 1: *Questionnaire responses. Responses are collected using a 5-point Likert scale (strongly disagree to strongly agree).*

	Question	Rating (-2 -1 0 1 2)
Q1	ChemoGraph helped navigate the enormous chemical space by interactively allowing user to control the direction of navigation	5 3
Q2	ChemoGraph helped discover paths between two compounds that are separated by multiple levels	2 3 3
Q3	ChemoGraph helped understand and visually compare compounds using different property models	1 3 4
Q4	ChemoGraph's ability to allow custom user models as input and visualize them on nodes is useful	8
Q5	ChemoGraph provides intuitive interactions to re-organize the nodes and helped quickly choose suitable chemical compounds/scaffolds	2 6
Q6	ChemoGraph helped in quickly creating a chemical series compared to other traditional ways	2 6
Q7	ChemoGraph tool is easy to use	1 7
Q8	How likely you are to use ChemoGraph tool in your research?	4 4

8. Discussion and Conclusion

We introduce ChemoGraph, a visual analysis tool for interactively exploring chemical space. We outline the importance of spatially exploring chemical space in the context of drug design. ChemoGraph formalizes chemical space as a hypergraph and defines the graph operations that allow users to navigate toward regions of interest. Based on the concept of scaffolds, we can represent entities within a few levels. Moreover, scaffold graphs have the potential to help medicinal chemists to design molecular series [LHW*20]. The case study suggests that ChemoGraph has well-defined practical applications in lead optimization and SAR series generation. We have also presented how users will be able to provide molecular data and automatically generate molecular series in contrast to manual enumeration of different possibilities. ChemoGraph can benefit other applications as well, such as model comparison.

Increasingly, scientists are training machine learning models to predict physical or biophysical properties of molecules (e.g., lipophilicity or binding affinity [GBWD*18, FSW*18]). However, with the increased usage of such models, deciding which models are valid for a specific task and interpreting the results for decision making remains a challenge [CDS20]. Thus, it is necessary to research novel methods to evaluate local differences between users' trained models on molecules. One unique challenge for working with molecular data is that without an obvious global representation, it is hard to understand how different models behave in chemical space. ChemoGraph uses a natural visual representation which makes it easy to evaluate the models and make comparisons.

8.1. Design Implications

The design of ChemoGraph highlights several advantages. First, it does not rely on static datasets. Though we can use a subset of chemical space as an initial input, ChemoGraph does not confine users to that dataset. With the help of generative models in the background, ChemoGraph improves the diversity of the explored compounds by generating novel scaffold classes on the fly. Second, compared to other chemical space exploration tools (e.g., Molpher [HŠVS14]), ChemoGraph presents a rich visual representation of chemical space by giving it a structure and encodes both molecular structures and numerical properties in the visual realm. This helps analysts to look at the multivariate nature of the compounds and explore chemical space in multiple dimensions.

8.2. Limitations and Future Work

We have identified several challenges for future investigation. Firstly, we would like to address the visual scalability of ChemoGraph. Our goal is not to visualize the entire chemical space, as our target user only focuses on local neighborhoods for their tasks. However, we want ChemoGraph to be scalable enough to handle increasing edge-crossings with a number of nodes. Also, we plan to add mechanisms to help users filter the space interactively, a feature that was highly requested by the experts in our study. For instance, adding a way to semantically filter the graph based on various properties (e.g., an unwanted chemical substructure or a range of values on a specific property) would enhance a user's focused exploration of local chemistries. We want to improve the visual encoding and evaluation of custom property prediction models. A few experts in our study were less satisfied with how ChemoGraph handles the evaluation of property models (Q3 in table 1). We would also like to investigate more useful ways of visualizing property models on the enumerated space for doing visual comparisons. Finally, we did not run a comparative study. Our initial case study was designed to understand the usefulness of ChemoGraph in common drug discovery tasks. Comparing ChemoGraph-aided tasks with baselines such as traditional by-hand enumerations would be valuable.

We believe that ChemoGraph provides a way to bridge the gap between computational and AI-driven chemistry and traditional expert-focused medicinal chemistry tasks. Hence, ChemoGraph opens the discussion of how AI complements human analysts during the exploratory sensemaking process and may help identify critical usability issues from the collaboration.

Acknowledgments

This research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357. This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the US DOE Office of Science and the National Nuclear Security Administration, the National Institute of Allergy and Infectious Diseases, National Institutes of Health Award Number P01AI165077 (AR), the National Science Foundation Award 2117896 and 2002082, and supported by the DOE through the National Virtual Biotechnology Laboratory, a consortium of DOE national laboratories focused on response to COVID-19, with funding from the Coronavirus CARES Act. We thank Laura Wolf for her help in editing the paper. We also acknowledge the anonymous reviewers for their time and valuable comments on improving the paper.

References

[BFS04] BÖHM H.-J., FLOHR A., STAHL M.: Scaffold hopping. *Drug discovery today: Technologies 1*, 3 (2004), 217–224. 4

[BM96] BEMIS G. W., MURCKO M. A.: The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry 39*, 15 (1996), 2887–2893. 4

[BMG96] BOHACEK R. S., MCMARTIN C., GUIDA W. C.: The art and practice of structure-based drug design: a molecular modeling perspective. *Medicinal research reviews 16*, 1 (1996), 3–50. 2, 4

[BTBC*21] BURCH M., TEN BRINKE K. B., CASTELLA A., PETERS G. K. S., SHTERIYANOV V., VLASVINKEL R.: Dynamic graph exploration by interactively linked node-link diagrams and matrix visualizations. *Visual Computing for Industry, Biomedicine, and Art 4*, 1 (2021), 1–14. 3

[CDS20] CLYDE A., DUAN X., STEVENS R.: Regression enrichment surfaces: a simple analysis technique for virtual drug screening models. *arXiv preprint arXiv:2006.01171* (2020). 10

[CGK*21] CLYDE A., GALANIE S., KNELLER D. W., MA H., BABUJI Y., BLAISZIK B., BRACE A., BRETTIN T., CHARD K., CHARD R., ET AL.: High-throughput virtual screening and validation of a sars-cov-2 main protease noncovalent inhibitor. *Journal of chemical information and modeling* (2021). 9

[CKS*21] CLYDE A., KALE B., SUN M., PAPKA M., RAMANATHAN A., STEVENS R.: Scaffold embeddings: Learning the structure spanned by chemical fragments, scaffolds and compounds. In *Workshop on Learning Meaningful Representation of Life* (2021). 2, 6

[CLMB16] COREL E., LOPEZ P., MÉHEUST R., BAPTESTE E.: Network-thinking: graphs to analyze microbial complexity and evolution. *Trends in Microbiology 24*, 3 (2016), 224–237. 2

[DRRD12] DÖRK M., RICHE N. H., RAMOS G., DUMAIS S.: Pivotpaths: Strolling through faceted information spaces. *IEEE transactions on visualization and computer graphics 18*, 12 (2012), 2709–2718. 2

[dSROFS19] DA SILVA ROCHA S. F., OLANDA C. G., FOKOUE H. H., SANT'ANNA C. M.: Virtual screening techniques in drug discovery: review and recent applications. *Current topics in medicinal chemistry 19*, 19 (2019), 1751–1767. 4

[FFKS21] FISCHER M. T., FRINGS A., KEIM D. A., SEEBACHER D.: Towards a survey on static and dynamic hypergraph visualizations. In *2021 IEEE Visualization Conference (VIS)* (2021), IEEE, pp. 81–85. 2, 3

[FSB*13] FIAUX P., SUN M., BRADEL L., NORTH C., RAMAKRISHNAN N., ENDERT A.: Bixplorer: Visual analytics with biclusters. *Computer 46*, 8 (2013), 90–94. 3

[FSSL21] FISCHER A., SMIESKO M., SELLNER M., LILL M. A.: Decision making in structure-based drug discovery: visual inspection of docking results. *Journal of Medicinal Chemistry 64*, 5 (2021), 2489–2500. 9

[FSW*18] FEINBERG E. N., SUR D., WU Z., HUSIC B. E., MAI H., LI Y., SUN S., YANG J., RAMSUNDAR B., PANDE V. S.: Potentialnet for molecular property prediction. *ACS central science 4*, 11 (2018), 1520–1530. 10

[GBWD*18] GÓMEZ-BOMBARELLI R., WEI J. N., DUVENAUD D., HERNÁNDEZ-LOBATO J. M., SÁNCHEZ-LENGELING B., SHEBERLA D., AGUILERA-IPARRAGUIRRE J., HIRZEL T. D., ADAMS R. P., ASPURU-GUZIK A.: Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science 4*, 2 (2018), 268–276. 10

[GJ79] GAREY M. R., JOHNSON D. S.: *Computers and intractability*, vol. 174. freeman San Francisco, 1979. 2

[GMH*18] GUPTA A., MÜLLER A. T., HUISMAN B. J., FUCHS J. A., SCHNEIDER P., SCHNEIDER G.: Generative recurrent networks for de novo drug design. *Molecular informatics 37*, 1-2 (2018), 1700111. 4

[Han76] HANSCH C.: Structure of medicinal chemistry. *Journal of Medicinal Chemistry 19*, 1 (1976), 1–6. 3

[HFM07] HENRY N., FEKETE J.-D., MCGUFFIN M. J.: Nodetrix: a hybrid visualization of social networks. *IEEE transactions on visualization and computer graphics 13*, 6 (2007), 1302–1309. 3

[HSB17] HU Y., STUMPFE D., BAJORATH J.: Recent advances in scaffold hopping: miniperspective. *Journal of medicinal chemistry 60*, 4 (2017), 1238–1246. 2

[HSBW11] HEINRICH J., SEIFERT R., BURCH M., WEISKOPF D.: Bicluster viewer: a visualization tool for analyzing gene expression data. In *International Symposium on Visual Computing* (2011), Springer, pp. 641–652. 3

[HŠVS14] HOKSZA D., ŠKODA P., VORŠILÁK M., SVOZIL D.: Molpher: a software framework for systematic chemical space exploration. *Journal of cheminformatics 6*, 1 (2014), 1–13. 3, 10

[ID09] INSELBERG A., DIMSDALE B.: Parallel coordinates. *Human-Machine Interactive Systems* (2009), 199–233. 5

[JLH*19] JIA X., LYNCH A., HUANG Y., DANIELSON M., LANG'AT I., MILDER A., RUBY A. E., WANG H., FRIEDLER S. A., NORQUIST A. J., ET AL.: Anthropogenic biases in chemical reaction data hinder exploratory inorganic synthesis. *Nature 573*, 7773 (2019), 251–255. 4

[KKC*04] KAPUSHESKY M., KEMMEREN P., CULHANE A. C., DURINCK S., IHMELS J., KÖRNER C., KULL M., TORRENTE A., SARKANS U., VILO J., ET AL.: Expression profiler: next generation—an online platform for analysis of microarray data. *Nucleic acids research 32*, suppl_2 (2004), W465–W470. 3

[KLG*21] KNELLER D. W., LI H., GALANIE S., PHILLIPS G., LABBÉ A., WEISS K. L., ZHANG Q., ARNOULD M. A., CLYDE A., MA H., ET AL.: Structural, electronic, and electrostatic determinants for inhibitor binding to subsites s1 and s2 in sars-cov-2 main protease. *Journal of medicinal chemistry 64*, 23 (2021), 17366–17383. 9

[KQR*19] KAKAR T., QIN X., RUNDENSTEINER E. A., HARRISON L., SAHOO S. K., DE S.: Diva: Exploration and validation of hypothesized drug-drug interactions. In *Computer Graphics Forum* (2019), vol. 38, Wiley Online Library, pp. 95–106. 3

[KSP21] KALE B., SUN M., PAPKA M. E.: Direct ordering: A direct manipulation based ordering technique. In *Workshop on Exploratory Search and Interactive Data Analytics* (2021). 6

[KTB*16] KIM S., THIESSEN P. A., BOLTON E. E., CHEN J., FU G., GINDULYTE A., HAN L., HE J., HE S., SHOEMAKER B. A., ET AL.: Pubchem substance and compound databases. *Nucleic acids research 44*, D1 (2016), D1202–D1213. 2

[Lan13] LANDRUM G.: Rdkit documentation. *Release 1*, 1-79 (2013), 4. 6

[LHW*20] LAI J., HU J., WANG Y., ZHOU X., LI Y., ZHANG L., LIU Z.: Privileged scaffold analysis of natural products with deep learning-based indication prediction model. *Molecular Informatics 39*, 11 (2020), 2000057. 10

[LWB*19] LYU J., WANG S., BALIUS T. E., SINGH I., LEVIT A., MOROZ Y. S., O'MEARA M. J., CHE T., ALGAA E., TOLMACHOVA K., ET AL.: Ultra-large library docking for discovering new chemotypes. *Nature 566*, 7743 (2019), 224–229. 4

[MFB21] MEYERS J., FABIAN B., BROWN N.: De novo molecular design and generative models. *Drug Discovery Today 26*, 11 (2021), 2707–2715. 9

[MFMMG*08] MEDINA-FRANCO J. L., MARTÍNEZ-MAYORGA K., GIULIANOTTI M. A., HOUGHTEN R. A., PINILLA C.: Visualization of the chemical space in drug discovery. *Current Computer-Aided Drug Design 4*, 4 (2008), 322–333. 3

[NMF17] NAVEJA J. J., MEDINA-FRANCO J. L.: Chemmaps: Towards an approach for visualizing the chemical space based on adaptive satellite compounds. *F1000Research 6* (2017). 2, 3

[NMF19] NAVEJA J. J., MEDINA-FRANCO J. L.: Finding constellations in chemical space through core analysis. *Frontiers in chemistry* (2019), 510. 2, 3

[NMSL19] NOBRE C., MEYER M., STREIT M., LEX A.: The state of the art in visualizing multivariate networks. In *Computer Graphics Forum* (2019), vol. 38, Wiley Online Library, pp. 807–832. 2

[ORO*15] OSOLODKIN D. I., RADCHENKO E. V., ORLOV A. A., VORONKOV A. E., PALYULIN V. A., ZEFIROV N. S.: Progress in visual representations of chemical space. *Expert opinion on drug discovery 10*, 9 (2015), 959–973. 2, 3

[PHE*17] PIENTA R., HOHMAN F., ENDERT A., TAMERSOY A., ROUNDY K., GATES C., NAVATHE S., CHAU D. H.: Vigor: interactive visual exploration of graph query results. *IEEE transactions on visualization and computer graphics 24*, 1 (2017), 215–225. 2

[PLS*14] PARTL C., LEX A., STREIT M., STROBELT H., WASSERMANN A.-M., PFISTER H., SCHMALSTIEG D.: Contour: data-driven exploration of multi-relational datasets for drug discovery. *IEEE transactions on visualization and computer graphics 20*, 12 (2014), 1883–1892. 3

[Rey15] REYMOND J.-L.: The chemical space project. *Accounts of Chemical Research 48*, 3 (2015), 722–730. 3, 4

[RVDBR12] RUDDIGKEIT L., VAN DEURSEN R., BLUM L. C., REYMOND J.-L.: Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of chemical information and modeling 52*, 11 (2012), 2864–2875. 2

[SBNR14] SUN M., BRADEL L., NORTH C. L., RAMAKRISHNAN N.: The role of interactive biclusters in sensemaking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2014), pp. 1559–1562. 3

[SC20] SCOTT O. B., CHAN A. W. E.: ScaffoldGraph: an open-source library for the generation and analysis of molecular scaffold networks and scaffold trees. *Bioinformatics* (03 2020). btaa219. 4

[SER*07] SCHUFFENHAUER A., ERTL P., ROGGO S., WETZEL S., KOCH M. A., WALDMANN H.: The scaffold tree- visualization of the scaffold universe by hierarchical scaffold classification. *Journal of chemical information and modeling 47*, 1 (2007), 47–58. 3

[SFEJ15] SOLTESZOVA V., FOSCATO M., ELIASSON S. H., JENSEN V. R.: Evolution inspector: Interactive visual analysis for evolutionary molecular design. In *2015 IEEE Conference on Visual Analytics Science and Technology (VAST)* (2015), IEEE, pp. 219–220. 3

[SFvKR15] SANDER T., FREYSS J., VON KORFF M., RUFENER C.: Datawarrior: an open-source program for chemistry aware data visualization and analysis. *Journal of chemical information and modeling 55*, 2 (2015), 460–473. 3

[SGG*14] STREIT M., GRATZL S., GILLHOFER M., MAYR A., MITTERECKER A., HOCHREITER S.: Furby: fuzzy force-directed bicluster visualization. *BMC bioinformatics 15*, 6 (2014), 1–13. 3

[SGL08] STASKO J., GÖRG C., LIU Z.: Jigsaw: supporting investigative analysis through interactive visualization. *Information visualization 7*, 2 (2008), 118–132. 3, 5

[SJUS08] SCHULZ H.-J., JOHN M., UNGER A., SCHUMANN H.: Visual analysis of bipartite biological networks. In *Eurographics Workshop on Visual Computing for Biomedicine* (2008). 3

[SK21] SHRIVASTAVA A. D., KELL D. B.: Fragnet, a contrastive learning-based transformer model for clustering, interpreting, visualizing, and navigating chemical space. *Molecules 26*, 7 (2021), 2065. 3

[SLAG18] SANCHEZ-LENGELING B., ASPURU-GUZIK A.: Inverse molecular design using machine learning: Generative models for matter engineering. *Science 361*, 6400 (2018), 360–365. 4

[SMNR15] SUN M., MI P., NORTH C., RAMAKRISHNAN N.: Biset: Semantic edge bundling with biclusters for sensemaking. *IEEE transactions on visualization and computer graphics 22*, 1 (2015), 310–319. 3, 5, 6

[SMS*17] SAHU S., MHEDHBI A., SALIHOGLU S., LIN J., ÖZSU M. T.: The ubiquity of large graphs and surprising challenges of graph processing. *Proceedings of the VLDB Endowment 11*, 4 (2017), 420–431. 2

[SNGS99] SCHNEIDER G., NEIDHART W., GILLER T., SCHMID G.: "scaffold-hopping" by topological pharmacophore search: a contribution to virtual screening. *Angewandte Chemie International Edition 38*, 19 (1999), 2894–2896. 2

[SNK*14] SUSHKO Y., NOVOTARSKYI S., KÖRNER R., VOGT J., ABDELAZIZ A., TETKO I. V.: Prediction-driven matched molecular pairs to interpret qsars and aid the molecular optimization process. *Journal of Cheminformatics 6*, 1 (2014), 1–18. 3

[SNR14] SUN M., NORTH C., RAMAKRISHNAN N.: A five-level design framework for bicluster visualizations. *IEEE transactions on visualization and computer graphics 20*, 12 (2014), 1713–1722. 3

[STS09] SCHNEIDER P., TANRIKULU Y., SCHNEIDER G.: Self-organizing maps in drug discovery: compound library design, scaffold-hopping, repurposing. *Current medicinal chemistry 16*, 3 (2009), 258–266. 2

[Sun16] SUN M.: *Visual Analytics with Biclusters: Exploring Coordinated Relationships in Context*. PhD thesis, Virginia Tech, 2016. 3

[SUS*20] SABANDO M. V., ULBRICH P., SELZER M., BYŠKA J., MIČAN J., PONZONI I., SOTO A. J., GANUZA M. L., KOZLÍKOVÁ B.: Chemva: interactive visual analysis of chemical compound similarity in virtual screening. *IEEE Transactions on Visualization and Computer Graphics 27*, 2 (2020), 891–901. 3

[SZW*18] SUN M., ZHAO J., WU H., LUTHER K., NORTH C., RAMAKRISHNAN N.: The effect of edge bundling and seriation on sensemaking of biclusters in bipartite graphs. *IEEE transactions on visualization and computer graphics 25*, 10 (2018), 2983–2998. 3, 6

[UM18] USLU T., MEHLER A.: Polyviz: a visualization system for a special kind of multipartite graphs. In *Proceedings of the IEEE VIS* (2018). 3

[VBP*19] VALDIVIA P., BUONO P., PLAISANT C., DUFOURNAUD N., FEKETE J.-D.: Analyzing dynamic hypergraphs with parallel aggregated ordered hypergraph visualization. *IEEE transactions on visualization and computer graphics 27*, 1 (2019), 1–13. 3

[Wei88] WEININGER D.: Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences 28*, 1 (1988), 31–36. 7

[WLWB10] WAWER M., LOUNKINE E., WASSERMANN A. M., BAJORATH J.: Data structures and computational tools for the extraction of sar information from large compound sets. *Drug Discovery Today 15*, 15-16 (2010), 630–639. 3

[WQT*21] WOO H.-M., QIAN X., TAN L., JHA S., ALEXANDER F. J., DOUGHERTY E. R., YOON B.-J.: Optimal decision making in high-throughput virtual screening pipelines. *arXiv preprint arXiv:2109.11683* (2021). 4

[WSM*18] WU H., SUN M., MI P., TATTI N., NORTH C., RAMAKRISHNAN N.: Interactive discovery of coordinated relationship chains with maximum entropy models. *ACM Transactions on Knowledge Discovery from Data (TKDD) 12*, 1 (2018), 1–34. 3

[ZSCC17] ZHAO J., SUN M., CHEN F., CHIU P.: Bidots: Visual exploration of weighted biclusters. *IEEE transactions on visualization and computer graphics 24*, 1 (2017), 195–204. 5

[ZSCC19] ZHAO J., SUN M., CHEN F., CHIU P.: Missbin: Visual analysis of missing links in bipartite networks. In *2019 IEEE Visualization Conference (VIS)* (2019), IEEE, pp. 71–75. 3

[ZSCC20] ZHAO J., SUN M., CHEN F., CHIU P.: Understanding missing links in bipartite networks with missbin. *IEEE Transactions on Visualization and Computer Graphics 28*, 6 (2020), 2457–2469. 3

[ZSYN15] ZHANG H., SUN M., YAO D., NORTH C.: Visualizing traffic causality for analyzing network anomalies. In *Proceedings of the 2015 ACM International Workshop on Security and Privacy Analytics* (2015), pp. 37–42. 5