ForVis - Doc4

Nieroda, Pala, Wojakowski

1 Introduction

The team's task was to extend the functionality of the existing ForVis service by adding methods for visualizing DIMACS files containing logical formulas. Representing these formulas using various techniques proves to be a useful tool in the process of formula reduction by graphically illustrating relationships between variables and clauses.

The project began with the concept of a determinant that represents the relationship between the frequency of a variable's occurrence within an entire formula and the length of the clause in which it appears.

During the project, a **heatmap** was implemented to visualize the computed relationships using colors. Additionally, an algorithm was developed to reduce the size of the final chart to prevent overloading users' web browsers.

Another visualization concept involving **hypergraphs** was explored. This method effectively depicts the membership of variables within clauses. However, the implementation process was halted due to the incompatibility of required libraries with the Angular framework version on which the service is built.

Finally, optimization work was carried out to improve the performance of the new visualization. Future development directions for the system were also suggested.

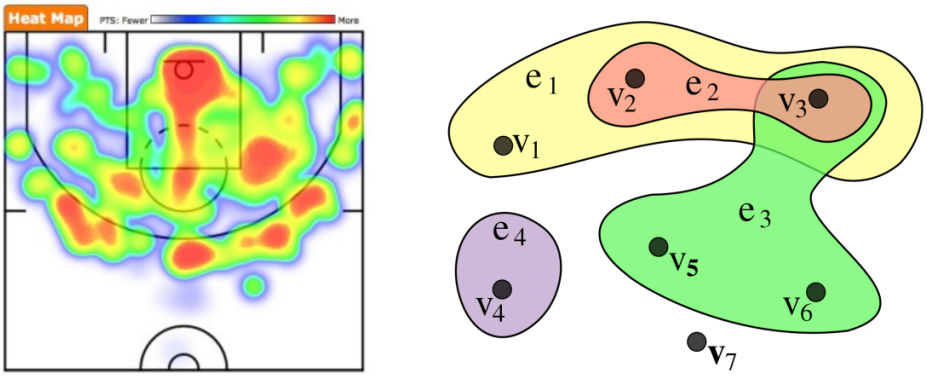

Figure 1

2 Heatmap

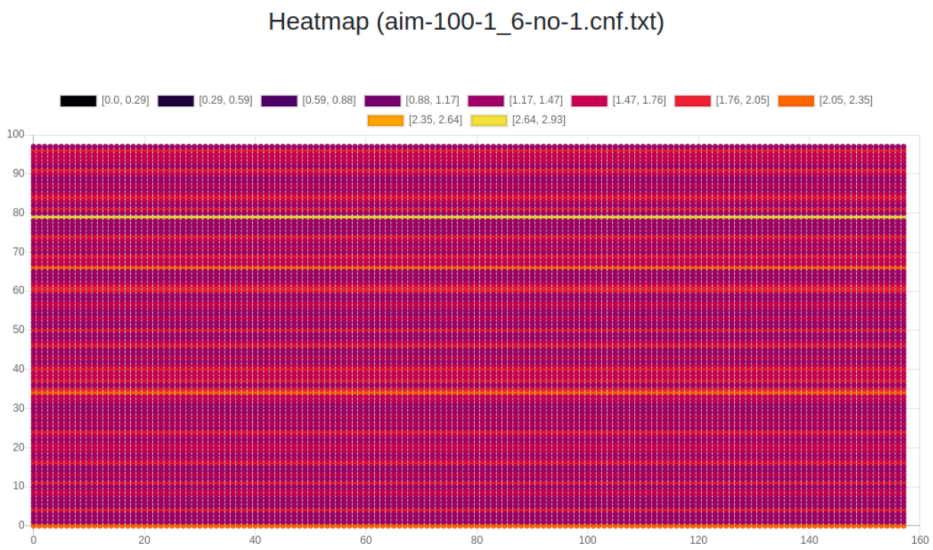

Figure 2: Sample heatmap generated by ForVis

The **heatmap** represents formulas as a pseudo-3D graph where the third dimension is expressed through color intensity. The **X-axis** represents the clause number, and the **Y-axis** represents variables indexed accordingly. For each two-dimensional point, a determinant ζ is calculated.

A **higher determinant value** is represented with **warmer colors**, while **lower values** are displayed with **cooler colors**.

2.1 Determinant Calculation (ζ)

The determinant is defined by the following formula:

$$\zeta = \frac{\text{count}(x)}{\text{length}(c)} \tag{1}$$

Where:

- x – logical variable appearing in the formula
- c – clause appearing in the formula
- **count** – function counting occurrences of a variable across the entire formula
- **length** – function counting the number of variables in a given clause

For a simple CNF clause:

```
p cnf 5 5
1 2 4 0
2 3 5 0
1 2 3 5 0
```

```
2 4 0
1 2 5 0
```

A matrix is created with dimensions (number of variables) $\times$ (number of clauses), where:

- 1 denotes the presence of a variable in a clause
- 0 denotes its absence
- -1 denotes its occurrence in a negated form

	X1	X2	X3	X4	X5
C1	1	1	0	1	0
C2	0	1	1	0	1
C3	1	1	1	0	1
C4	0	1	0	1	0
C5	1	1	0	0	1

Table 1: Example matrix representation of CNF clauses

For the point $(X1, C1)$:

- Count$(X1) = 3$ (number of occurrences in the entire formula)
- Length$(C1) = 3$ (number of variables in $C1$)
- ζ value at $(X1, C1) = 1$

After computing values for each point, they are scaled and assigned colors to generate the heatmap. As a result of the calculations for the proposed formula, we obtain the following heatmap:

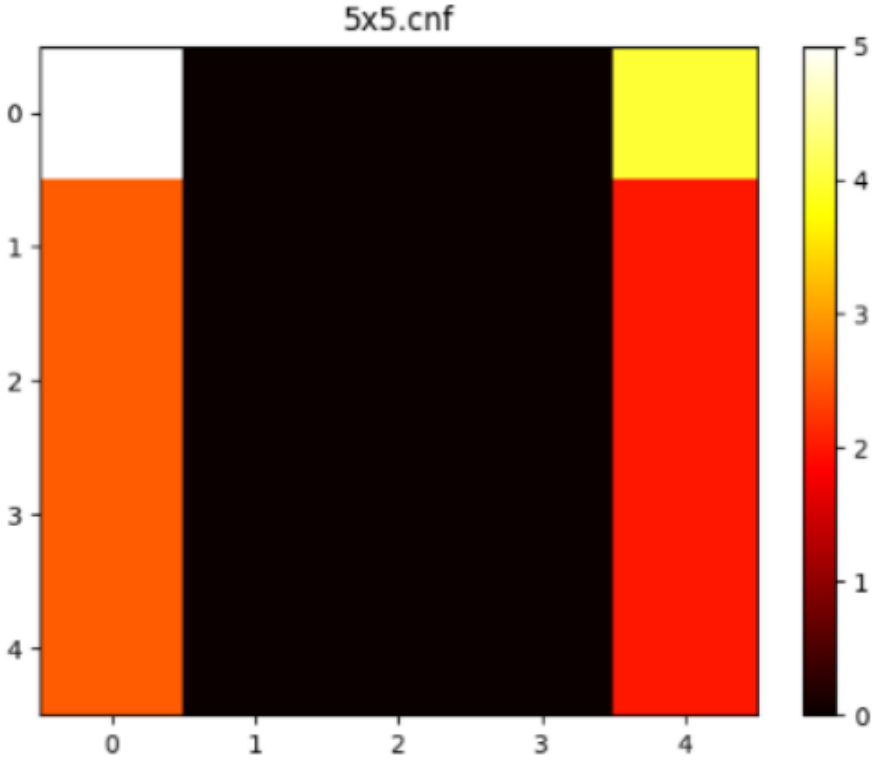

Figure 3: Sample heatmap generated by ForVis

3 Interpretation of the Visualization

3.1 Warmer (hotter) areas

- A frequently occurring variable suggests that its assignment impacts a large number of formulas, making it crucial for formula satisfiability.
- If a frequently appearing variable is in a **small clause**, then the satisfiability of that clause depends on many others.

3.2 Cooler (colder) areas

- A variable that appears rarely has less effect on the formula.
- If the clause containing the variable depends on many other variables, then its importance in the local context diminishes.

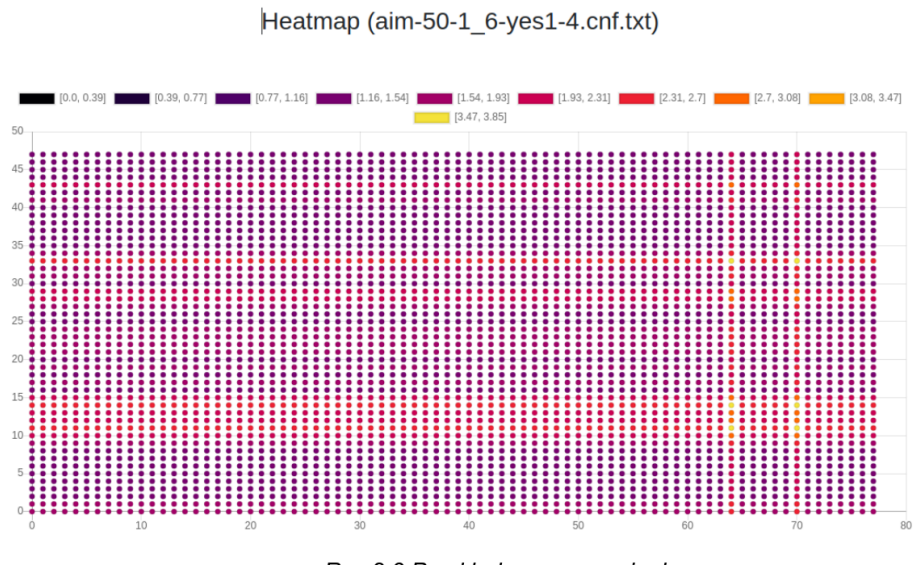

Figure 4: Sample heatmap generated by ForVis

4 HyperGraph Representation

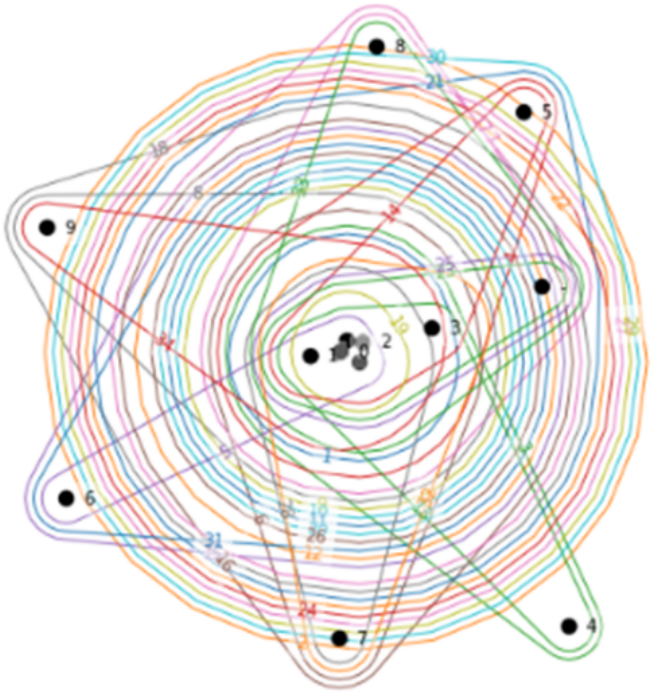

Figure 5: Sample hypergraph generated locally

The **hypergraph** visualization represents logical formulas as **sets of relationships between clauses and variables**.

- **Variables** are shown as **nodes**.

- **Clauses** are depicted as **surrounding areas**.
- **A node inside an area** represents a variable **belonging to a clause**.

4.1 Interpretation:

- Variables **in the center** occur **in many clauses**.
- Greater **distance from the center** means a variable appears **less frequently** in the formula.
- Variables **on the outer edges** can be **easily reduced** since they influence fewer clauses.

5 Integration Issues

During hypergraph integration, the team encountered problems due to **the lack of visualization libraries** that support hypergraph representations in the system. The issue was primarily caused by **outdated versions of Angular and VisJS** used in the project.

Alternative solutions, such as generating hypergraphs in **Python**, were tested but were unsuccessful due to compatibility issues. The **HyperX library** lacked support for saving graphs as files, making it difficult to render outside of **Jupyter Notebooks**.

6 Suggestions for Future Development

6.1 Update Angular to version 13/14

- The current **Angular 5** version is outdated and unsupported.
- Most modern visualization libraries are **not compatible** with this version.
- A **full version update** is required.

6.2 Create a Git Repository

- Currently, updating the code requires **manual operations** with Docker containers.
- **Using Git** would simplify updates and collaboration.
- **GitLab** is available via the university's license.

6.3 Database Structure Modification

- Currently, visualizations are stored as JSON objects within SQL tables.
- A **new table structure** with optimized fields should be implemented.

6.4 Separate Development and Production Configurations

- Currently, **only production settings exist**, requiring a **full system restart** for any changes.
- A **developer-friendly configuration** should be introduced for easier testing and debugging.