1 Map coloring problem

The objective of the map coloring problem is to create a map where neighboring countries are assigned distinct colors. In the realm of SAT problems, we inquire whether there exists a truth value assignment to the propositional variables present in the formula Φ under which the formula Φ evaluates to "true". Satisfying assignments are called "models" of Φ and form the solutions of the respective instance of SAT.

Following the notation from the paper, the state of the map is represented by a matrix $\mathbf{S} = \left\{s_1^1, s_1^2, \ldots, s_n^M\right\}$, $i \in [1, n]$, $j \in [1, M]$ with $s_i^j = 1$ denoting the region i being colored with the color j. Given the borders between all regions on the map (represented by a border adjacency matrix B with the elements $b_{ii'} = 1$ denoting region i and region i' sharing a border and $b_{ii'} = 0$ otherwise), the goal is to color all regions by distinct colors such that no two bordering regions have the same color.

Example - 4 countries, 2 colors

Let's say that the map of the four countries that we are looking to color is that of Czechia, Slovakia, Austria and Hungary. For $n = 4$ regions and $M = 2$ colors there are $N = nM = 8$ states, so the state of the map is represented by a matrix $\mathbf{S} = \left\{s_1^1, s_1^2, s_2^1, s_2^2, s_3^1, s_3^2, s_4^1, s_4^2,\right\}$, where Czechia is represented by the two states $\left\{s_1^1, s_1^2\right\}$, Slovakia is represented by $\left\{s_2^1, s_2^2\right\}$, and so on. The constraints of the problem are:

1. Each region has to be colored:

$$\begin{aligned}\varphi_1 &= \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{M} \left(s_i^j \right) \right) \\ &= \left(s_1^1 \vee s_1^2\right) \wedge \left(s_2^1 \vee s_2^2\right) \wedge \left(s_3^1 \vee s_3^2\right) \wedge \left(s_4^1 \vee s_4^2\right). \end{aligned} \tag{1}$$

2. A region cannot be two distinct colors at the same time:

$$\begin{aligned}\varphi_2 &= \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{M} \bigwedge_{j' \neq j}^{M} \left(\neg \left(s_i^j \wedge s_i^{j'} \right) \right) \\ &= \left(\neg \left(s_1^1 \wedge s_1^2\right)\right) \wedge \left(\neg \left(s_2^1 \wedge s_2^2\right)\right) \wedge \left(\neg \left(s_3^1 \wedge s_3^2\right)\right) \wedge \left(\neg \left(s_4^1 \wedge s_4^2\right)\right) \\ &= \left(\neg s_1^1 \vee \neg s_1^2\right) \wedge \left(\neg s_2^1 \vee \neg s_2^2\right) \wedge \left(\neg s_3^1 \vee \neg s_3^2\right) \wedge \left(\neg s_4^1 \vee \neg s_4^2\right). \end{aligned} \tag{2}$$

3. Regions that share a border should have a different color:

$$
\begin{aligned}
\varphi_3 =& \bigwedge_{i=1}^{n} \bigwedge_{\{i' \mid b_{ii'}=1\}} \bigwedge_{j=1}^{M} \left(\neg \left(s_i^j \wedge s_{i'}^j \right) \right) \\
=& \left(\neg s_1^1 \vee \neg s_2^1\right) \wedge \left(\neg s_1^1 \vee \neg s_3^1\right) \wedge \left(\neg s_2^1 \vee \neg s_3^1\right) \wedge \left(\neg s_2^1 \vee \neg s_4^1\right) \wedge \left(\neg s_3^1 \vee \neg s_4^1\right) \\
& \wedge \left(\neg s_1^2 \vee \neg s_2^2\right) \wedge \left(\neg s_1^2 \vee \neg s_3^2\right) \wedge \left(\neg s_2^2 \vee \neg s_3^2\right) \wedge \left(\neg s_2^2 \vee \neg s_4^2\right) \wedge \left(\neg s_3^2 \vee \neg s_4^2\right),
\end{aligned} \tag{3}
$$

where we used the adjacency matrix B for Czechia, Slovakia, Austria and Hungary:

$$
B = \begin{bmatrix} b_{11} & \cdots & b_{14} \\ \vdots & \ddots & \vdots \\ b_{41} & \cdots & b_{44} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}. \tag{4}
$$

Taking the sets of clauses (1), (2), and (3) together gives the formula for the map coloring problem of $n = 4$ regions (Czechia, Slovakia, Austria and Hungary) with $M = 2$ colors:

$$
\begin{aligned}
\Phi =& \varphi_1 \wedge \varphi_2 \wedge \varphi_3 \\
=& \left(s_1^1 \vee s_1^2\right) \wedge \left(s_2^1 \vee s_2^2\right) \wedge \left(s_3^1 \vee s_3^2\right) \wedge \left(s_4^1 \vee s_4^2\right) \\
& \wedge \left(\neg s_1^1 \vee \neg s_1^2\right) \wedge \left(\neg s_2^1 \vee \neg s_2^2\right) \wedge \left(\neg s_3^1 \vee \neg s_3^2\right) \wedge \left(\neg s_4^1 \vee \neg s_4^2\right) \\
& \wedge \left(\neg s_1^1 \vee \neg s_2^1\right) \wedge \left(\neg s_1^1 \vee \neg s_3^1\right) \wedge \left(\neg s_2^1 \vee \neg s_3^1\right) \wedge \left(\neg s_2^1 \vee \neg s_4^1\right) \wedge \left(\neg s_3^1 \vee \neg s_4^1\right) \\
& \wedge \left(\neg s_1^2 \vee \neg s_2^2\right) \wedge \left(\neg s_1^2 \vee \neg s_3^2\right) \wedge \left(\neg s_2^2 \vee \neg s_3^2\right) \wedge \left(\neg s_2^2 \vee \neg s_4^2\right) \wedge \left(\neg s_3^2 \vee \neg s_4^2\right).
\end{aligned} \tag{5}
$$

Using the Abdullah method to translate a logical formula to weights of the Hopfield network

According to Abdullah (1992) determining the states $\mathbf{S}$ that will satisfy Φ is equivalent to a combinatorial minimization of the cost function $E_{\neg\Phi}$ of the inconsistency $\neg\Phi$. The inconsistency $\neg\Phi$ for (5) is

$$
\begin{aligned}
\neg\Phi =& \left(\neg s_1^1 \wedge \neg s_1^2\right) \vee \left(\neg s_2^1 \wedge \neg s_2^2\right) \vee \left(\neg s_3^1 \wedge \neg s_3^2\right) \vee \left(\neg s_4^1 \wedge \neg s_4^2\right) \\
& \vee \left(s_1^1 \wedge s_1^2\right) \vee \left(s_2^1 \wedge s_2^2\right) \vee \left(s_3^1 \wedge s_3^2\right) \vee \left(s_4^1 \wedge s_4^2\right) \\
& \vee \left(s_1^1 \wedge s_2^1\right) \vee \left(s_1^1 \wedge s_3^1\right) \vee \left(s_2^1 \wedge s_3^1\right) \vee \left(s_2^1 \wedge s_4^1\right) \vee \left(s_3^1 \wedge s_4^1\right) \\
& \vee \left(s_1^2 \wedge s_2^2\right) \vee \left(s_1^2 \wedge s_3^2\right) \vee \left(s_2^2 \wedge s_3^2\right) \vee \left(s_2^2 \wedge s_4^2\right) \vee \left(s_3^2 \wedge s_4^2\right).
\end{aligned} \tag{6}
$$

The literals s_i and $\neg s_i$ in (6) are mapped to the terms $\frac{1}{2}\left(1 + s_i\right)$ and $\frac{1}{2}\left(1 - s_i\right)$, respectively. The entire disjunction of conjuctions is subsequently turned into a sum of products of such terms. In the example given above, the corresponding cost function $E_{\neg\Phi}$ takes the form

$$
\begin{aligned}
E_{\neg\Phi} =& \frac{1}{4}\left(1-s_1^1\right)\left(1-s_1^2\right)+\frac{1}{4}\left(1-s_2^1\right)\left(1-s_2^2\right)+\frac{1}{4}\left(1-s_3^1\right)\left(1-s_3^2\right) \\
&+\frac{1}{4}\left(1-s_4^1\right)\left(1-s_4^2\right)+\frac{1}{4}\left(1+s_1^1\right)\left(1+s_1^2\right)+\frac{1}{4}\left(1+s_2^1\right)\left(1+s_2^2\right) \\
&+\frac{1}{4}\left(1+s_3^1\right)\left(1+s_3^2\right)+\frac{1}{4}\left(1+s_4^1\right)\left(1+s_4^2\right)+\frac{1}{4}\left(1+s_1^1\right)\left(1+s_2^1\right) \\
&+\frac{1}{4}\left(1+s_1^1\right)\left(1+s_3^1\right)+\frac{1}{4}\left(1+s_1^1\right)\left(1+s_4^1\right)+\frac{1}{4}\left(1+s_2^1\right)\left(1+s_4^1\right) \\
&+\frac{1}{4}\left(1+s_3^1\right)\left(1+s_4^1\right)+\frac{1}{4}\left(1+s_1^2\right)\left(1+s_2^2\right)+\frac{1}{4}\left(1+s_1^2\right)\left(1+s_3^2\right) \\
&+\frac{1}{4}\left(1+s_1^2\right)\left(1+s_4^2\right)\frac{1}{4}\left(1+s_2^2\right)\left(1+s_4^2\right)+\frac{1}{4}\left(1+s_3^2\right)\left(1+s_4^2\right) \\
=& -\frac{1}{2}\left(-s_1^1 s_1^2-s_2^1 s_2^2-s_3^1 s_3^2-s_4^1 s_4^2-\frac{1}{2}s_1^1 s_2^1-\frac{1}{2}s_1^1 s_3^1-\frac{1}{2}s_1^1 s_4^1\right. \\
&\left.-\frac{1}{2}s_2^1 s_4^1-\frac{1}{2}s_3^1 s_4^1-\frac{1}{2}s_1^2 s_2^2-\frac{1}{2}s_1^2 s_3^2-\frac{1}{2}s_1^2 s_4^2-\frac{1}{2}s_2^2 s_4^2-\frac{1}{2}s_3^2 s_4^2\right) \\
&-\left(-\frac{3}{4}s_1^1-\frac{3}{4}s_1^2-\frac{1}{2}s_2^1-\frac{1}{2}s_2^2-\frac{1}{2}s_3^1-\frac{1}{2}s_3^2-\frac{3}{4}s_4^1-\frac{3}{4}s_4^2\right)-\left(-\frac{9}{2}\right).
\end{aligned}
\tag{7}
$$

Comparing (7) with the Hopfield network energy function term by term:

$$
E\left(t\right)=-\frac{1}{2}\sum_i^N\sum_j^N W_{ij}^{(2)}s_i\left(t\right)s_j\left(t\right)-\sum_i^N W_i^{(1)}s_i\left(t\right)-c, \tag{8}
$$

we get that

$$
\begin{aligned}
\mathbf{W}^{(2)} &= \begin{bmatrix} W_{11} & \cdots & W_{18} \\ \vdots & \ddots & \vdots \\ W_{81} & \cdots & W_{88} \end{bmatrix} \\
&= \begin{bmatrix}
0 & -\frac{1}{2} & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 \\
-\frac{1}{2} & 0 & 0 & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 & -\frac{1}{4} \\
-\frac{1}{4} & 0 & 0 & -\frac{1}{2} & 0 & 0 & -\frac{1}{4} & 0 \\
0 & -\frac{1}{4} & -\frac{1}{2} & 0 & 0 & 0 & 0 & -\frac{1}{4} \\
-\frac{1}{4} & 0 & 0 & 0 & 0 & -\frac{1}{2} & -\frac{1}{4} & 0 \\
0 & -\frac{1}{4} & 0 & 0 & -\frac{1}{2} & 0 & 0 & -\frac{1}{4} \\
-\frac{1}{4} & 0 & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 & 0 & -\frac{1}{2} \\
0 & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 & -\frac{1}{4} & -\frac{1}{2} & 0
\end{bmatrix},
\end{aligned}
$$

where we assumed symmetrical weight connections, i.e. $W_{ij}=W_{ji}$,

$$
\mathbf{W}^{(1)}=\begin{bmatrix} -\frac{3}{4} & -\frac{3}{4} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{3}{4} & -\frac{3}{4} \end{bmatrix},
$$

and

$$c = -\frac{9}{2}.$$

Weighing the constraints by border length

The small map of Czechia, Slovakia, Austria and Hungary requires at least 3 colors such that no two bordering regions have the same color. As was mentioned in the paper, if we have just two colors at our disposal, we can still find an optimal coloring scheme to obtain a map as comprehensible as possible, if we put additional weights ω_i on the different set of constraints φ_i in (5):

$$\Phi = \omega_1\varphi_1 \wedge \omega_2\varphi_2 \wedge \omega_3\varphi_3. \tag{9}$$

So for example using the weights $\omega = \left[1, 1, b^L_{ii'}\right]$, where the border constraints φ_3 are weighted by a normalized border adjacency matrix B^L (with the elements $b^L_{ii'}$ denoting the border length between countries i and i'), we would get that Czechia and Hungary would be colored with one color, and Slovakia and Austria would be colored with the second color.

References

Abdullah, W. A. T. W. (1992). Logic programming on a neural network. *International Journal of Intelligent Systems*, 7(6):513–519.