# E.A.6.6 (Edge Colouring)

## 1.1 Modellazione

Dato un grafo non diretto $G = (V, E)$ siano
- $\mathcal{C} = \{1, 2, 3\}$
- $X = \left\{ X_{u,v}^c \mid (u,v) \in E \land c \in \mathcal{C} \land u < v \right\}$ l'insieme di variabili t.c.
  - $X_{u,v}^c$ è vera se l'arco $(u,v) \in E$ ha colore $c$

$$\phi = \phi_{\text{almeno\_un\_colore\_per\_arco}} \land \phi_{\text{al\_più\_un\_colore\_per\_arco}} \land \phi_{\text{triangoli}}$$

$$\phi_{\text{almeno\_un\_colore\_per\_arco}} = \bigwedge_{\substack{(u,v) \in E \\ u < v}} \left( \bigvee_{c \in \mathcal{C}} X_{u,v}^c \right)$$

$$\phi_{\text{al\_più\_un\_colore\_per\_arco}} = \bigwedge_{\substack{(u,v) \in E \\ c_1, c_2 \in \mathcal{C} \\ u < v \\ c_1 < c_2}} \left( X_{u,v}^{c_1} \to \neg X_{u,v}^{c_2} \right)$$

$$\phi_{\text{triangoli}} = \bigwedge_{\substack{u,v,w \in V \\ (u,v),(v,w),(u,w) \in E \\ c \in \mathcal{C} \\ u < v < w}} \left( X_{u,v}^c \land X_{v,w}^c \to \neg X_{u,w}^c \right)$$

## 1.2 Istanziazione

### 1.2.1 Variabili

Dato $G = (V, E)$ come in Figura 1.1 si ha

$\bar{}\quad X = \big\{$

$\qquad X_{A,E}^1, X_{A,E}^2, X_{A,E}^3, X_{A,H}^1, X_{A,H}^2, X_{A,H}^3,$

$\qquad X_{A,I}^1, X_{A,I}^2, X_{A,I}^3, X_{A,S}^1, X_{A,S}^2, X_{A,S}^3,$

$\qquad X_{B,C}^1, X_{B,C}^2, X_{B,C}^3, X_{B,\ G2}^1, X_{B,\ G2}^2, X_{B,\ G2}^3,$

$\qquad X_{B,I}^1, X_{B,I}^2, X_{B,I}^3, X_{B,J}^1, X_{B,J}^2, X_{B,J}^3,$

$\qquad X_{B,S}^1, X_{B,S}^2, X_{B,S}^3, X_{C,D}^1, X_{C,D}^2, X_{C,D}^3,$

$\qquad X_{C,\ G2}^1, X_{C,\ G2}^2, X_{C,\ G2}^3, X_{C,S}^1, X_{C,S}^2, X_{C,S}^3,$

$\qquad X_{D,E}^1, X_{D,E}^2, X_{D,E}^3, X_{D,S}^1, X_{D,S}^2, X_{D,S}^3,$

$\qquad X_{E,\ G1}^1, X_{E,\ G1}^2, X_{E,\ G1}^3, X_{E,H}^1, X_{E,H}^2, X_{E,H}^3,$

$\qquad X_{G1,H}^1, X_{G1,H}^2, X_{G1,H}^3, X_{G2,J}^1, X_{G2,J}^2, X_{G2,J}^3,$

$\qquad X_{H,I}^1, X_{H,I}^2, X_{H,I}^3,$

$\quad \big\}$

### 1.2.2 Vincoli

$\qquad \varphi_{\text{almeno\_un\_colore\_per\_arco}} = \big\{$

$\qquad (X_{A,E}^1 \vee X_{A,E}^2 \vee X_{A,E}^3) \wedge (X_{A,H}^1 \vee X_{A,H}^2 \vee X_{A,H}^3) \wedge$

$\qquad (X_{A,I}^1 \vee X_{A,I}^2 \vee X_{A,I}^3) \wedge (X_{A,S}^1 \vee X_{A,S}^2 \vee X_{A,S}^3) \wedge$

$\qquad (X_{B,C}^1 \vee X_{B,C}^2 \vee X_{B,C}^3) \wedge (X_{B,\ G2}^1 \vee X_{B,\ G2}^2 \vee X_{B,\ G2}^3) \wedge$

$\qquad (X_{B,I}^1 \vee X_{B,I}^2 \vee X_{B,I}^3) \wedge (X_{B,J}^1 \vee X_{B,J}^2 \vee X_{B,J}^3) \wedge$

$\qquad (X_{B,S}^1 \vee X_{B,S}^2 \vee X_{B,S}^3) \wedge (X_{C,D}^1 \vee X_{C,D}^2 \vee X_{C,D}^3) \wedge$

$\qquad (X_{C,\ G2}^1 \vee X_{C,\ G2}^2 \vee X_{C,\ G2}^3) \wedge (X_{C,S}^1 \vee X_{C,S}^2 \vee X_{C,S}^3) \wedge$

$\qquad (X_{D,E}^1 \vee X_{D,E}^2 \vee X_{D,E}^3) \wedge (X_{D,S}^1 \vee X_{D,S}^2 \vee X_{D,S}^3) \wedge$

$\qquad (X_{E,\ G1}^1 \vee X_{E,\ G1}^2 \vee X_{E,\ G1}^3) \wedge (X_{E,H}^1 \vee X_{E,H}^2 \vee X_{E,H}^3) \wedge$

$\qquad (X_{G1,H}^1 \vee X_{G1,H}^2 \vee X_{G1,H}^3) \wedge (X_{G2,J}^1 \vee X_{G2,J}^2 \vee X_{G2,J}^3) \wedge$

$\qquad (X_{H,I}^1 \vee X_{H,I}^2 \vee X_{H,I}^3)$

$\qquad \big\}$

$$\varphi_{\text{al\_più\_un\_colore\_per\_arco}} = \Big\{$$

$$\left(\neg X_{A,E}^1 \vee \neg X_{A,E}^2\right) \wedge \left(\neg X_{A,E}^1 \vee \neg X_{A,E}^3\right) \wedge \left(\neg X_{A,E}^2 \vee \neg X_{A,E}^3\right) \wedge$$

$$\left(\neg X_{A,H}^1 \vee \neg X_{A,H}^2\right) \wedge \left(\neg X_{A,H}^1 \vee \neg X_{A,H}^3\right) \wedge \left(\neg X_{A,H}^2 \vee \neg X_{A,H}^3\right) \wedge$$

$$\left(\neg X_{A,I}^1 \vee \neg X_{A,I}^2\right) \wedge \left(\neg X_{A,I}^1 \vee \neg X_{A,I}^3\right) \wedge \left(\neg X_{A,I}^2 \vee \neg X_{A,I}^3\right) \wedge$$

$$\left(\neg X_{A,S}^1 \vee \neg X_{A,S}^2\right) \wedge \left(\neg X_{A,S}^1 \vee \neg X_{A,S}^3\right) \wedge \left(\neg X_{A,S}^2 \vee \neg X_{A,S}^3\right) \wedge$$

$$\left(\neg X_{B,C}^1 \vee \neg X_{B,C}^2\right) \wedge \left(\neg X_{B,C}^1 \vee \neg X_{B,C}^3\right) \wedge \left(\neg X_{B,C}^2 \vee \neg X_{B,C}^3\right) \wedge$$

$$\left(\neg X_{B,\ G2}^1 \vee \neg X_{B,\ G2}^2\right) \wedge \left(\neg X_{B,\ G2}^1 \vee \neg X_{B,\ G2}^3\right) \wedge \left(\neg X_{B,\ G2}^2 \vee \neg X_{B,\ G2}^3\right) \wedge$$

$$\left(\neg X_{B,I}^1 \vee \neg X_{B,I}^2\right) \wedge \left(\neg X_{B,I}^1 \vee \neg X_{B,I}^3\right) \wedge \left(\neg X_{B,I}^2 \vee \neg X_{B,I}^3\right) \wedge$$

$$\left(\neg X_{B,J}^1 \vee \neg X_{B,J}^2\right) \wedge \left(\neg X_{B,J}^1 \vee \neg X_{B,J}^3\right) \wedge \left(\neg X_{B,J}^2 \vee \neg X_{B,J}^3\right) \wedge$$

$$\left(\neg X_{B,S}^1 \vee \neg X_{B,S}^2\right) \wedge \left(\neg X_{B,S}^1 \vee \neg X_{B,S}^3\right) \wedge \left(\neg X_{B,S}^2 \vee \neg X_{B,S}^3\right) \wedge$$

$$\left(\neg X_{C,D}^1 \vee \neg X_{C,D}^2\right) \wedge \left(\neg X_{C,D}^1 \vee \neg X_{C,D}^3\right) \wedge \left(\neg X_{C,D}^2 \vee \neg X_{C,D}^3\right) \wedge$$

$$\left(\neg X_{C,\ G2}^1 \vee \neg X_{C,\ G2}^2\right) \wedge \left(\neg X_{C,\ G2}^1 \vee \neg X_{C,\ G2}^3\right) \wedge \left(\neg X_{C,\ G2}^2 \vee \neg X_{C,\ G2}^3\right) \wedge$$

$$\left(\neg X_{C,S}^1 \vee \neg X_{C,S}^2\right) \wedge \left(\neg X_{C,S}^1 \vee \neg X_{C,S}^3\right) \wedge \left(\neg X_{C,S}^2 \vee \neg X_{C,S}^3\right) \wedge$$

$$\left(\neg X_{D,E}^1 \vee \neg X_{D,E}^2\right) \wedge \left(\neg X_{D,E}^1 \vee \neg X_{D,E}^3\right) \wedge \left(\neg X_{D,E}^2 \vee \neg X_{D,E}^3\right) \wedge$$

$$\left(\neg X_{D,S}^1 \vee \neg X_{D,S}^2\right) \wedge \left(\neg X_{D,S}^1 \vee \neg X_{D,S}^3\right) \wedge \left(\neg X_{D,S}^2 \vee \neg X_{D,S}^3\right) \wedge$$

$$\left(\neg X_{E,\ G1}^1 \vee \neg X_{E,\ G1}^2\right) \wedge \left(\neg X_{E,\ G1}^1 \vee \neg X_{E,\ G1}^3\right) \wedge \left(\neg X_{E,\ G1}^2 \vee \neg X_{E,\ G1}^3\right) \wedge$$

$$\left(\neg X_{E,H}^1 \vee \neg X_{E,H}^2\right) \wedge \left(\neg X_{E,H}^1 \vee \neg X_{E,H}^3\right) \wedge \left(\neg X_{E,H}^2 \vee \neg X_{E,H}^3\right) \wedge$$

$$\left(\neg X_{G1,H}^1 \vee \neg X_{G1,H}^2\right) \wedge \left(\neg X_{G1,H}^1 \vee \neg X_{G1,H}^3\right) \wedge \left(\neg X_{G1,H}^2 \vee \neg X_{G1,H}^3\right) \wedge$$

$$\left(\neg X_{G2,J}^1 \vee \neg X_{G2,J}^2\right) \wedge \left(\neg X_{G2,J}^1 \vee \neg X_{G2,J}^3\right) \wedge \left(\neg X_{G2,J}^2 \vee \neg X_{G2,J}^3\right) \wedge$$

$$\left(\neg X_{H,I}^1 \vee \neg X_{H,I}^2\right) \wedge \left(\neg X_{H,I}^1 \vee \neg X_{H,I}^3\right) \wedge \left(\neg X_{H,I}^2 \vee \neg X_{H,I}^3\right)$$

$$\Big\}$$

$$\varphi_{\text{triangoli}} = \Big\{$$

$$\left(\neg X_{A,E}^1 \vee \neg X_{E,H}^1 \vee \neg X_{A,H}^1\right) \wedge \left(\neg X_{A,E}^2 \vee \neg X_{E,H}^2 \vee \neg X_{A,H}^2\right) \wedge$$

$$\left(\neg X_{A,E}^3 \vee \neg X_{E,H}^3 \vee \neg X_{A,H}^3\right) \wedge \left(\neg X_{A,H}^1 \vee \neg X_{H,I}^1 \vee \neg X_{A,I}^1\right) \wedge$$

$$\left(\neg X_{A,H}^2 \vee \neg X_{H,I}^2 \vee \neg X_{A,I}^2\right) \wedge \left(\neg X_{A,H}^3 \vee \neg X_{H,I}^3 \vee \neg X_{A,I}^3\right) \wedge$$

$$\left(\neg X_{B,C}^1 \vee \neg X_{C,\ G2}^1 \vee \neg X_{B,\ G2}^1\right) \wedge \left(\neg X_{B,C}^2 \vee \neg X_{C,\ G2}^2 \vee \neg X_{B,\ G2}^2\right) \wedge$$

$$\left(\neg X_{B,C}^3 \vee \neg X_{C,\ G2}^3 \vee \neg X_{B,\ G2}^3\right) \wedge \left(\neg X_{B,C}^1 \vee \neg X_{C,S}^1 \vee \neg X_{B,S}^1\right) \wedge$$

$$\left(\neg X_{B,C}^2 \vee \neg X_{C,S}^2 \vee \neg X_{B,S}^2\right) \wedge \left(\neg X_{B,C}^3 \vee \neg X_{C,S}^3 \vee \neg X_{B,S}^3\right) \wedge$$

$$\left(\neg X_{B,\ G2}^1 \vee \neg X_{G2,J}^1 \vee \neg X_{B,J}^1\right) \wedge \left(\neg X_{B,\ G2}^2 \vee \neg X_{G2,J}^2 \vee \neg X_{B,J}^2\right) \wedge$$

$$\left(\neg X_{B,\ G2}^3 \vee \neg X_{G2,J}^3 \vee \neg X_{B,J}^3\right) \wedge \left(\neg X_{C,D}^1 \vee \neg X_{D,S}^1 \vee \neg X_{C,S}^1\right) \wedge$$

$$\left(\neg X_{C,D}^2 \vee \neg X_{D,S}^2 \vee \neg X_{C,S}^2\right) \wedge \left(\neg X_{C,D}^3 \vee \neg X_{D,S}^3 \vee \neg X_{C,S}^3\right) \wedge$$

$$\left(\neg X_{E,\ G1}^1 \vee \neg X_{G1,H}^1 \vee \neg X_{E,H}^1\right) \wedge \left(\neg X_{E,\ G1}^2 \vee \neg X_{G1,H}^2 \vee \neg X_{E,H}^2\right) \wedge$$

$$\left(\neg X_{E,\ G1}^3 \vee \neg X_{G1,H}^3 \vee \neg X_{E,H}^3\right) \wedge$$

}

# 1.3 Codifica in SATCodec

### 1.3.1 EdgeColouringToSAT

```java
import it.uniroma1.di.tmancini.utils.*;
import it.uniroma1.di.tmancini.teaching.ai.SATCodec.*;
import java.util.*;

public class EdgeColouringToSAT {
  public static void main(String args[]) {

    var AEdges = new ArrayList<Integer>();
    AEdges.add(4);
    AEdges.add(7);
    AEdges.add(8);
    AEdges.add(10);

    var BEdges = new ArrayList<Integer>();
    BEdges.add(2);
    BEdges.add(6);
    BEdges.add(8);
    BEdges.add(9);
    BEdges.add(10);

    var CEdges = new ArrayList<Integer>();
    CEdges.add(3);
    CEdges.add(6);
    CEdges.add(10);

    var DEdges = new ArrayList<Integer>();
    DEdges.add(4);
    DEdges.add(10);

    var EEdges = new ArrayList<Integer>();
    EEdges.add(5);
    EEdges.add(7);

    var G1Edges = new ArrayList<Integer>();
    G1Edges.add(7);

    var G2Edges = new ArrayList<Integer>();
    G2Edges.add(9);

    var HEdges = new ArrayList<Integer>();
    HEdges.add(8);

    var IEdges = new ArrayList<Integer>();
    var JEdges = new ArrayList<Integer>();
    var SEdges = new ArrayList<Integer>();

    var edges = new ArrayList<ArrayList<Integer>>();
    edges.add(AEdges);
    edges.add(BEdges);
```

```java
        edges.add(CEdges);
        edges.add(DEdges);
        edges.add(EEdges);
        edges.add(G1Edges);
        edges.add(G2Edges);
        edges.add(HEdges);
        edges.add(IEdges);
        edges.add(JEdges);
        edges.add(SEdges);

        // A = 0
        // B = 1
        // C = 2
        // D = 3
        // E = 4
        // G1 = 5
        // G2 = 6
        // H = 7
        // I = 8
        // J = 9
        // S = 10

        var colorsRange = new IntRange("colors", 1, 3);
        var edgesRange = new IntRange("edges", 0, 10);

        var encoder = new SATEncoder("EdgeColuring", "edge-
colouring.cnf");
        encoder.defineFamilyOfVariables("X", edgesRange, edgesRange,
colorsRange);

        // Almeno un colore per arco
        for (Integer u : edgesRange.values()) {
          for (Integer v : edges.get(u)) {
            if (u < v) {
              for (Integer c : colorsRange.values()) {
                encoder.addToClause("X", u, v, c);
              }
              encoder.endClause();
            }
          }
        }

        // Al più un colore per arco
        for (Integer u : edgesRange.values()) {
          for (Integer v : edges.get(u)) {
            if (u < v) {
              for (Integer c1 : colorsRange.values()) {
                for (int c2 = c1 + 1; c2 ≤ 3; c2++) {
                  encoder.addNegToClause("X", u, v, c1);
                  encoder.addNegToClause("X", u, v, c2);
                  encoder.endClause();
                }
              }
```

```
        }
      }
    }

    // Triangoli
    for (Integer u : edgesRange.values()) {
      for (Integer v : edgesRange.values()) {
        for (Integer w : edgesRange.values()) {
          if (u < v && v < w && edges.get(u).contains(v) &&
edges.get(v).contains(w)
              && edges.get(u).contains(w)) {
            for (Integer c : colorsRange.values()) {
              encoder.addNegToClause("X", u, v, c);
              encoder.addNegToClause("X", v, w, c);
              encoder.addNegToClause("X", u, w, c);
              encoder.endClause();
            }
          }
        }
      }
    }

    encoder.end();
  }
}
```

### 1.3.2 SATToEdgeColouring

```
import it.uniroma1.di.tmancini.utils.*;
import it.uniroma1.di.tmancini.teaching.ai.SATCodec.*;
import java.util.*;

public class SATToEdgeColouring {
  public static void main(String args[]) throws
java.io.IOException, java.io.FileNotFoundException {
    String[] variables = { "A", "B", "C", "D", "E", "G1", "G2",
"H", "I", "J", "S" };

    var decoder = new SATModelDecoder(args);
    decoder.run();

    int maxVar = decoder.getMaxVar();
    for (int i = 1; i ≤ maxVar; i++) {
      Boolean v_i = decoder.getModelValue(i);
      if (v_i == null || !v_i)
        continue;

      SATModelDecoder.Var variable = decoder.decodeVariable(i);
      int u = variable.getIndices().get(0);
      int v = variable.getIndices().get(1);
      int c = variable.getIndices().get(2);
      System.out.println(variables[u] + " → " + variables[v] +
" : " + c);
```
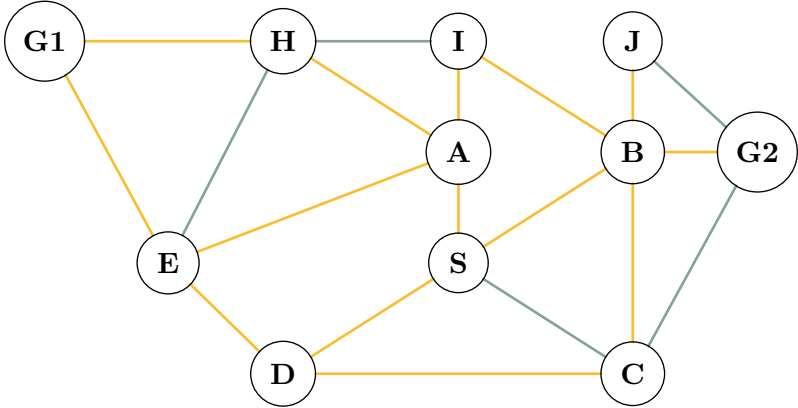
```
        }
        System.out.println("-- STOP HERE");
    }
}
```



Figura 1: soluzione generata da `picosat`