LISN
LABORATOIRE INTERDISCIPLINAIRE
DES SCIENCES DU NUMÉRIQUE

université
PARIS-SACLAY

cnrs

Artificial Intelligence

Cours8 - Explicabilité en IA

L3 - Informatique

Nadjib Lazaar
Ing - Phd - HDR - Professor - Paris-Saclay University - LISN - LaHDAK
lazaar@lisn.fr https://perso.lisn.upsaclay.fr/lazaar/
14/03/2025

Introduction à l'explicabilité

Qu'est-ce que l'explicabilité ?

- L'explicabilité en IA désigne la capacité d'un modèle à rendre ses décisions compréhensibles et interprétables par les utilisateurs

- **Pourquoi l'explicabilité est-elle essentielle ?**
  - Confiance des utilisateurs
  - Conformité aux réglementations
  - Amélioration et audit des modèles
  - Éviter les biais et les discriminations

Illustrations de l'explicabilité

Exemple 1 : Diagnostic médical assisté par IA

- Un modèle d'IA prédit la présence d'une maladie à partir d'images médicales
- Sans explicabilité : le médecin ne sait pas pourquoi l'IA classe un patient comme à risque
- Avec des outils comme SHAP ou Grad-CAM : on peut visualiser les zones de l'image qui influencent la prédiction, aidant ainsi à la validation du diagnostic
- **Bénéfices :**
  - Meilleure acceptation par les médecins
  - Conformité réglementaire
  - Réduction des erreurs médicales

Illustrations de l'explicabilité

Exemple 2 : Conduite autonome

- Un véhicule autonome décide de freiner brusquement
    - Sans explicabilité : le conducteur ne sait pas pourquoi l'IA a pris cette décision
    - Avec des méthodes d'explicabilité : on peut voir que l'IA a détecté un obstacle sur la route et a estimé qu'il fallait freiner
- **Bénéfices :**
    - Meilleure compréhension des décisions des véhicules autonomes
    - Amélioration de la confiance des utilisateurs
    - Possibilité d'ajuster les modèles pour éviter des freinages inutiles

Cadre réglementaire : AI-Act

L'AI-Act : principes et impacts

- Proposition de règlementation européenne sur l'IA présentée en avril 2021
- Adoption en 2024, application progressive jusqu'en 2026
- Classification des systèmes d'IA selon le niveau de risque (minimal, élevé, inacceptable)
- Exigences accrues en termes de transparence et d'explicabilité

Cadre réglementaire : AI-Act

Les 7 exigences de l'AI-Act

1. **Gestion des risques :** évaluation et réduction des risques liés aux systèmes d'IA

2. **Gouvernance des données :** assurance de la qualité et de l'intégrité des données d'entraînement

3. **Documentation et transparence :** obligation de décrire le fonctionnement et les limites des modèles

4. **Explicabilité et traçabilité :** suivi et interprétation des décisions prises par l'IA

5. **Surveillance humaine :** intervention humaine pour contrôler les décisions automatiques

6. **Robustesse et cybersécurité :** garantir la résistance aux attaques et erreurs

7. **Conformité éthique :** respect des droits fondamentaux et absence de discrimination

Cadre réglementaire : AI-Act

Comparaison avec le RGPD

Critère	RGPD	AI-Act
Objet	Protection des données personnelles	Réglementation des systèmes d'IA
Transparence	Droits des individus sur les données	Explicabilité et documentation des systèmes
Obligations	Consentement, droit à l'explication	Audit, gestion des risques

Méthodes d’explicabilité

IA hybride, IA Neuro-Symbolique…

L’IA symbolique, avec sa structure logique explicite et son raisonnement basé sur des règles, apporte un avantage majeur dans l’explicabilité. Elle permet de répondre directement aux questions de **pourquoi** et **comment** une décision a été prise, en suivant une logique claire et transparente

Méthodes d’explicabilité

Apport de l’IA symbolique dans l’explicabilité

- **Raisonnement explicite** : Les systèmes symboliques utilisent des règles logiques et des arbres de décision qui peuvent être suivis et interprétés facilement par des humains

- **Traçabilité accrue** : Chaque étape du raisonnement est justifiable et transparente

- **Simplicité** : Contrairement aux techniques d’apprentissage profond, les systèmes symboliques sont souvent plus simples et compréhensibles

- **Conformité éthique et juridique** : Les règles et les décisions peuvent être formalisées, ce qui facilite la validation selon les réglementations comme le RGPD ou l’AI-Act

Définition de l’incohérence dans un modèle IA

Incohérence

- Une incohérence survient lorsqu’un modèle d’IA prend des décisions contradictoires ou lorsque des prédictions ne respectent pas des contraintes logiques ou des règles internes attendues

- **Exemple :** Un modèle de diagnostic qui prédit à la fois “présence de maladie” et “absence de maladie” pour le même patient avec des données contradictoires

- **Impact :** L’incohérence perturbe la confiance dans le modèle et rend les décisions difficiles à justifier. Elle peut également enfreindre des normes légales ou éthiques, comme celles imposées par l’AI-Act ou le RGPD

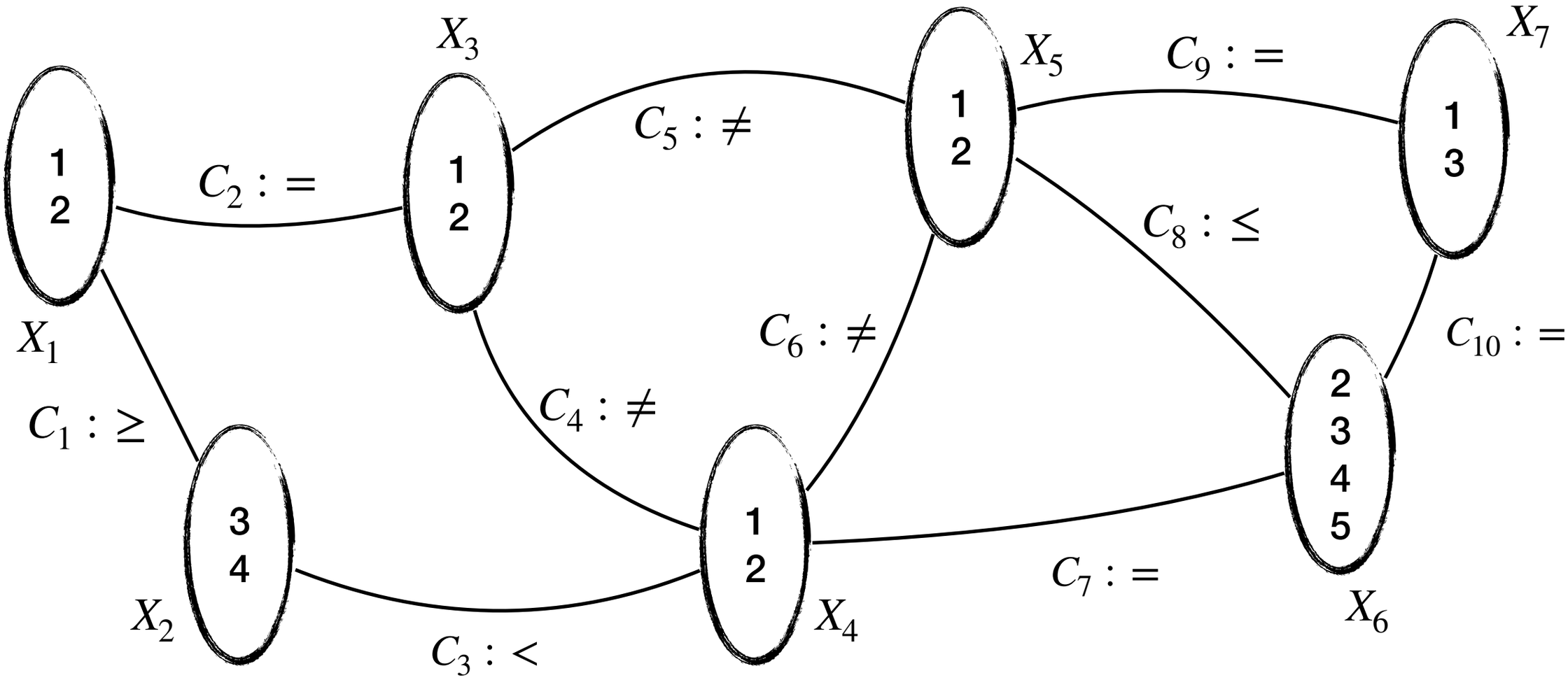
Explicabilité de l’incohérence en PPC

Unsatisfiabilité

- **Incohérence dans un modèle à contraintes** : Une incohérence survient lorsqu’un ensemble de contraintes ne peut pas être satisfait simultanément. Cela peut être dû à des relations contradictoires entre les variables ou à une situation où aucune solution ne respecte toutes les contraintes imposées

Explicabilité de l'incohérence en PPC

Unsatisfiabilité



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Minimal Unsatisfiable Subsets (MUS)

- Un **Minimal Unsatisfiable Subset (MUS)** est un sous-ensemble minimal d'un ensemble de contraintes qui est **insatisfiable**. Cela signifie que si l'on supprime une seule contrainte du MUS, l'ensemble restant devient satisfiable :

- Soit C un ensemble de contraintes tel que C est insatisfiable. Un sous-ensemble $M \subseteq C$ est un MUS si et seulement si :

  1. **Insatisfaisabilité** : M est insatisfiable

  2. **Minimalité** : Pour tout $c_i \in M$, l'ensemble $M \setminus \{c_i\}$ est satisfiable

Explicabilité de l'incohérence en PPC

Les Minimal Unsatisfiable Subsets (MUS)

- **Pourquoi les MUS sont importants ?**
  - **Diagnostic d'incohérence** : Identifier les contraintes responsables d'une incohérence dans un modèle
  - **Explication** : Permet de comprendre pourquoi un problème n'a pas de solution
  - **Correction** : Aide à suggérer des modifications minimales pour restaurer la cohérence
- **Propriétés des MUS**
  - **Insatisfaisabilité** : Un MUS ne peut pas être satisfait en l'état
  - **Minimalité** : Retirer une contrainte d'un MUS le rend satisfiable
  - **Non-uniqueness** : Un problème peut avoir plusieurs MUS différents

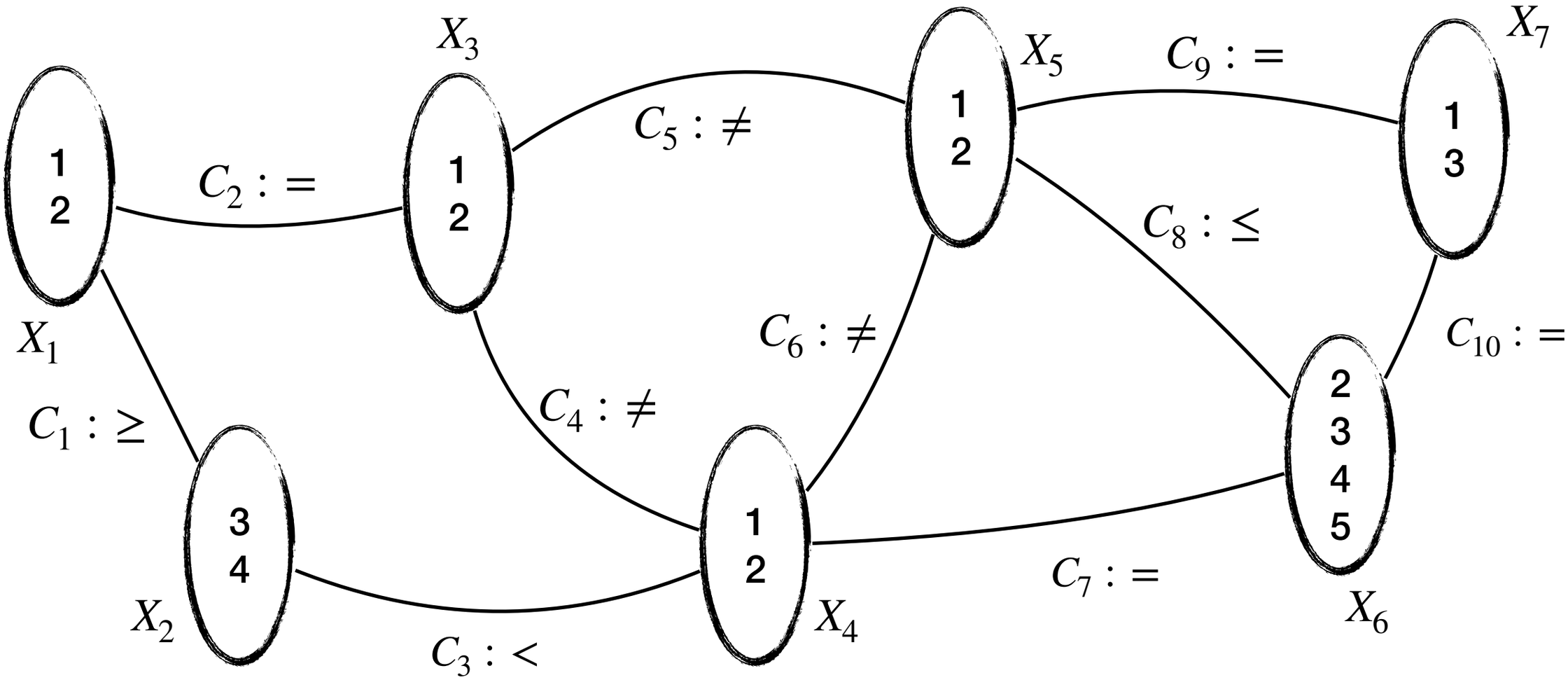
Explicabilité de l’incohérence en PPC

Les Minimal Unsatisfiable Subsets (MUS)

- **Exemple : Problème de Planification**
  - **Contexte** : Un employé ne peut pas être assigné à plusieurs tâches en même temps
  - Contraintes :
    - (C1) L’employé doit travailler sur le projet A de 9h à 11h
    - (C2) L’employé doit être disponible pour une réunion de 10h à 12h
    - (C3) Une tâche critique requiert sa présence de 10h30 à 11h30
- **Incohérence** : L’employé ne peut pas être à trois endroits en même temps
- **Quels sont les MUS dans ce cas ?**

Explicabilité de l'incohérence en PPC

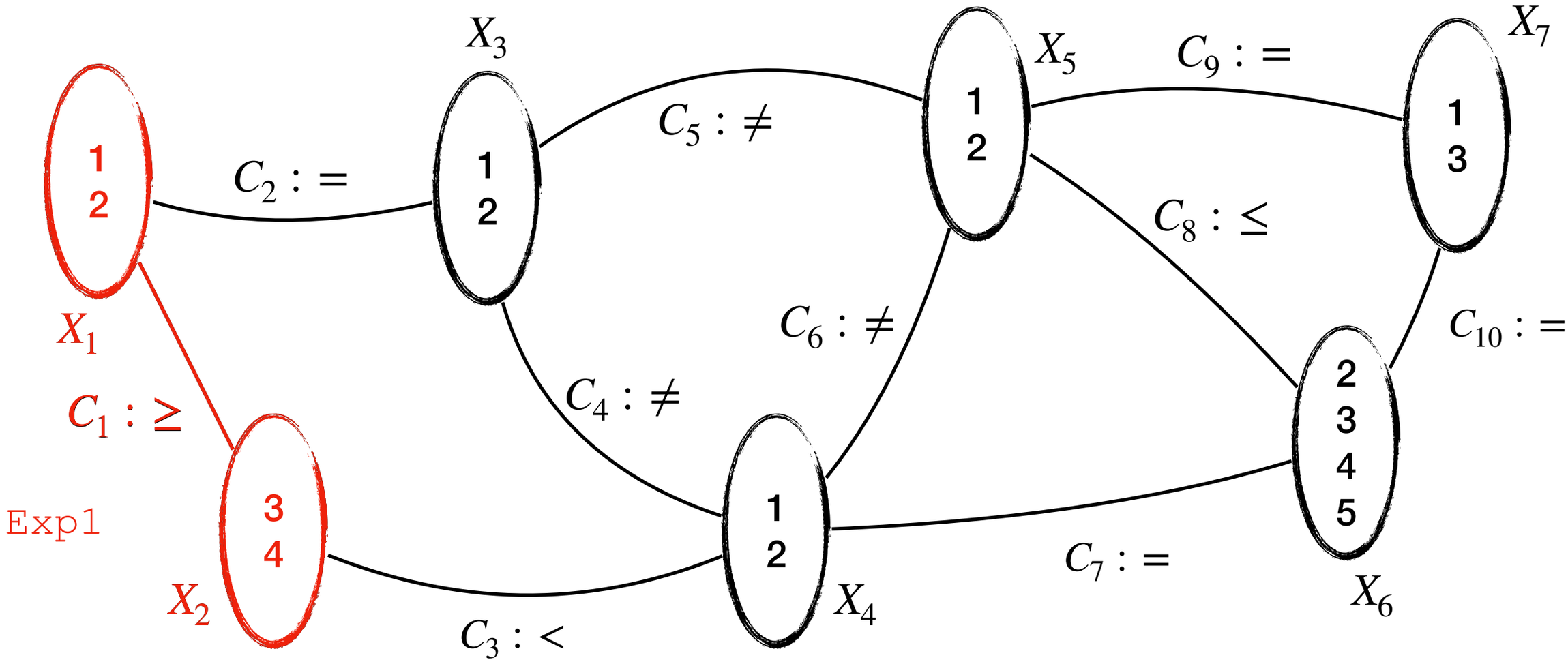
Les MUS



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

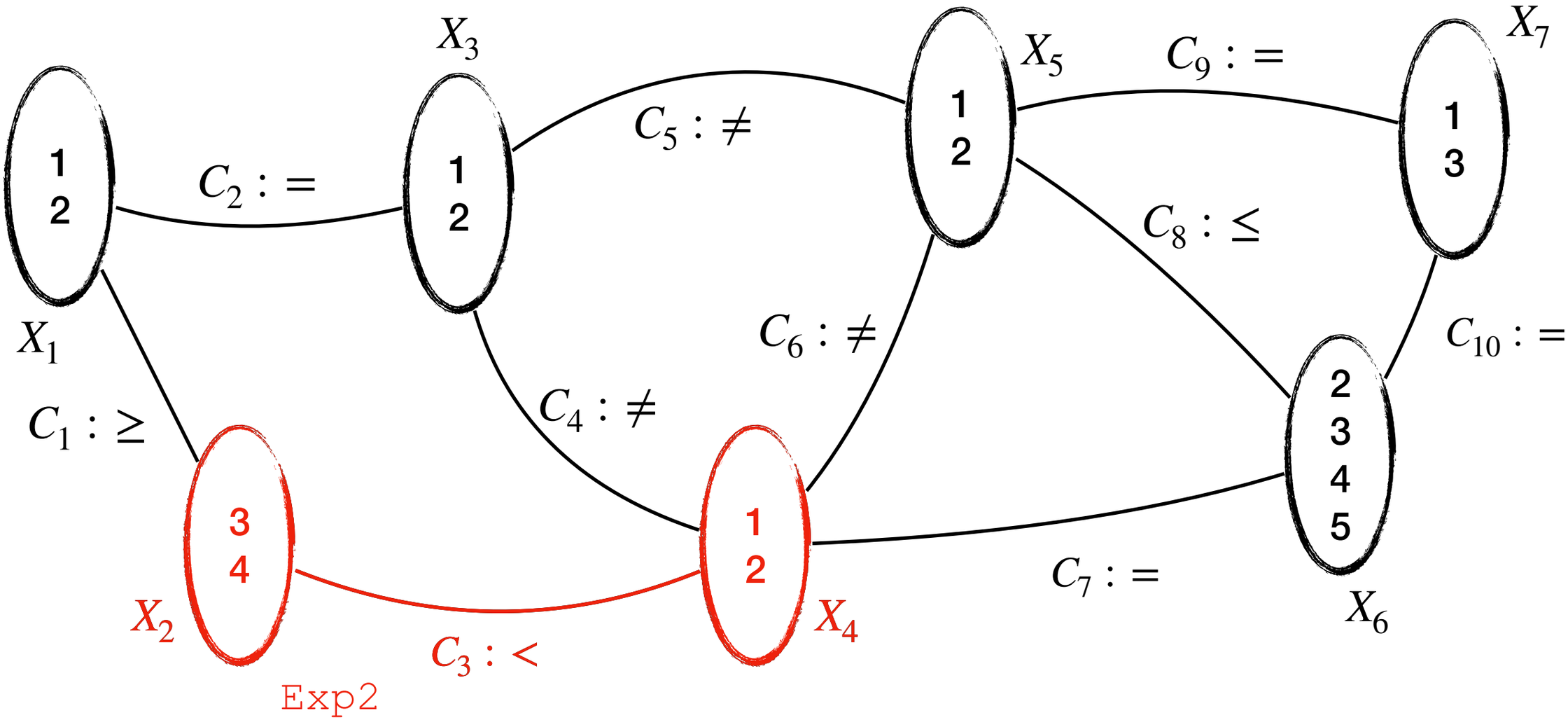
Les MUS



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

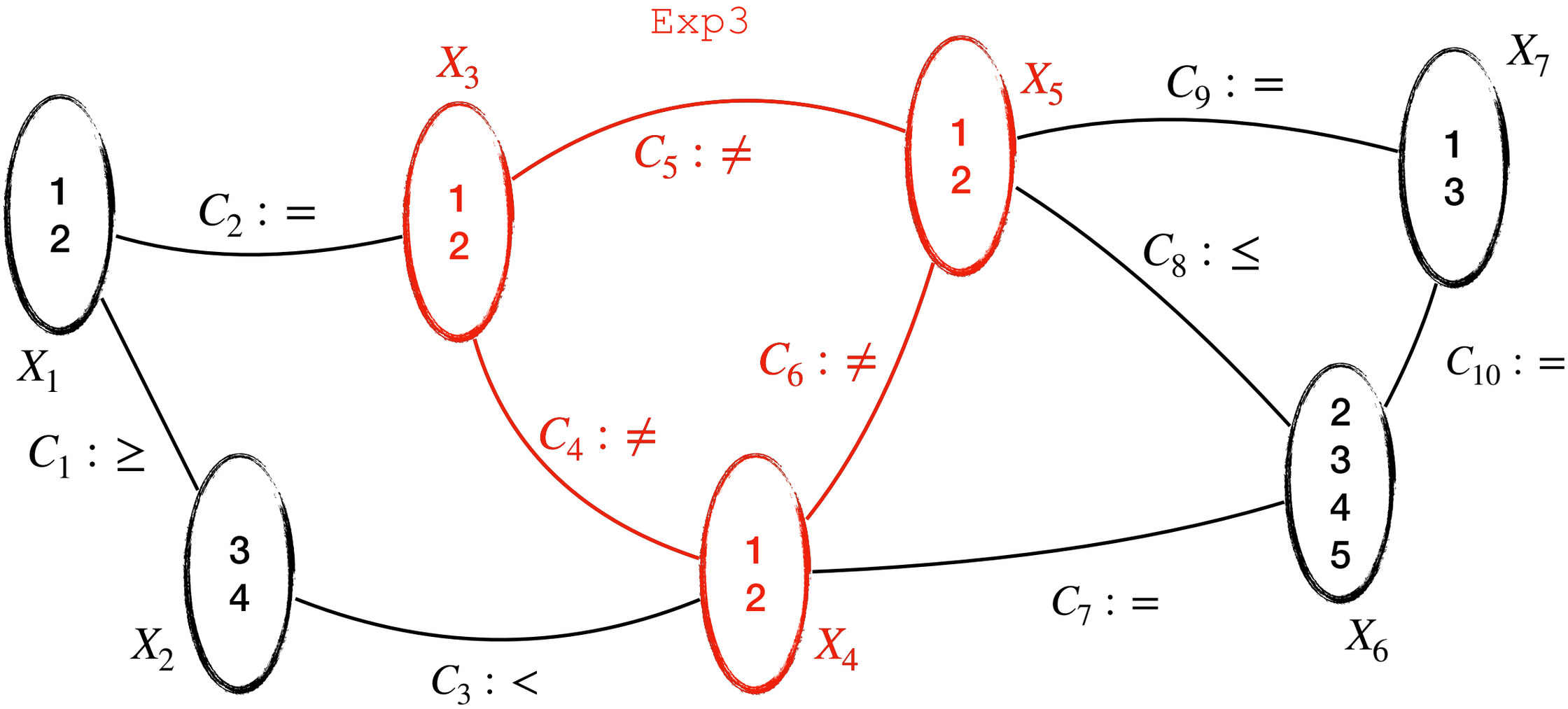
Les MUS



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

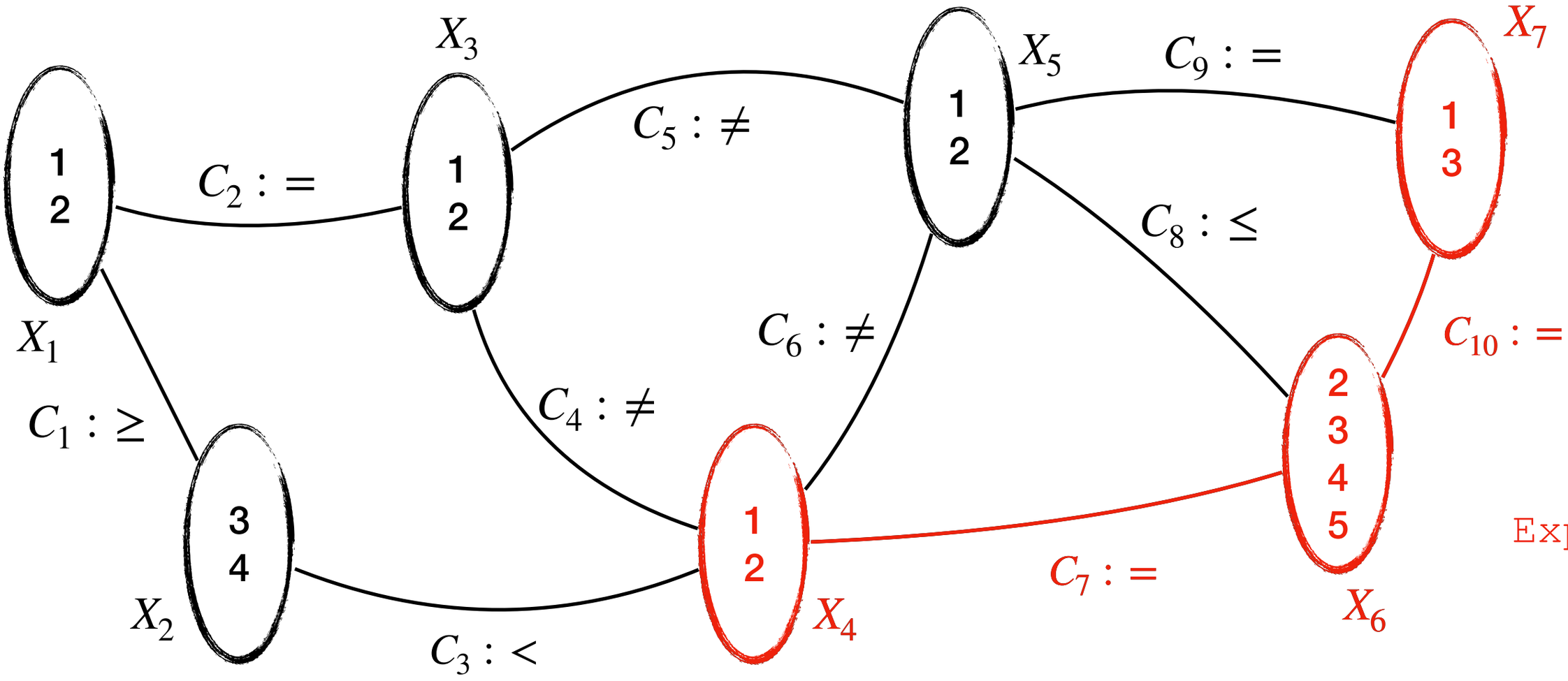
Les MUS



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les MUS

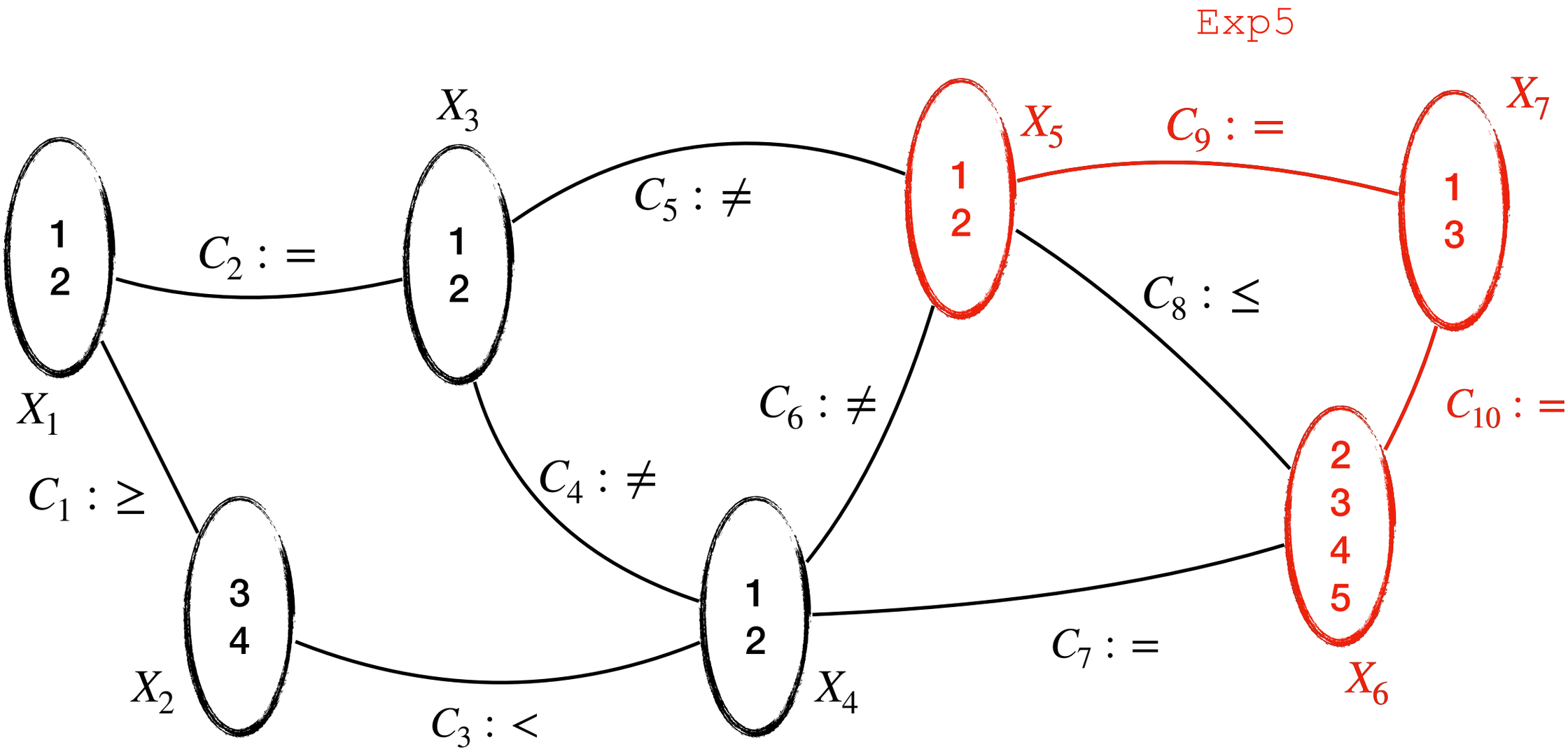


Exp4

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les MUS



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l’incohérence en PPC

Les MUS

- **Comment exploiter les MUS ?**

- Réviser le modèle : Identifier les contraintes à modifier

- Utiliser un MCS (Minimal Correction Subset) : Trouver les corrections minimales à apporter pour rendre le modèle satisfiable.

Explicabilité de l'incohérence en PPC

Les MCS

- Un Minimal Correction Subset (MCS) est un sous-ensemble minimal de contraintes qui, une fois supprimé d'un ensemble de contraintes insatisfiable, rend le problème satisfiable

- Soit C un ensemble de contraintes tel que C est insatisfiable. Un sous-ensemble $S \subseteq C$ est un Minimal Correction Subset (MCS) si et seulement si :

  - **Correction** : L'ensemble des contraintes restantes après suppression de S, soit $C \backslash S$, est **satisfiable**

  - **Minimalité** : Pour tout $S' \subset S$, l'ensemble $C \backslash S'$ **reste insatisfiable**, c'est-à-dire qu'aucune contrainte de S n'est superflue : $\forall S' \subset S, (C \backslash S')$ est insatisfiable.

Explicabilité de l'incohérence en PPC

Les MCS

- **Pourquoi les MCS sont importants ?**

  - **Correction d'incohérences :** Permet d'identifier les contraintes à modifier ou supprimer pour restaurer la cohérence

  - **Optimisation :** Trouver les ajustements minimaux nécessaires dans un système basé sur des contraintes

  - **Complément des MUS :** Un MCS est un complément à l'analyse des MUS, car retirer un MCS d'un ensemble insatisfiable laisse un **MSS (Minimum Satisfiable Subset)**

- **Propriétés des MCS**

  1. Correction minimale : Supprimer un MCS de C garantit que l'ensemble restant devient satisfiable

  2. Lien avec les MUS : Chaque MCS correspond à au moins un MUS dans l'ensemble des contraintes

  3. Non-unicité : Un problème peut avoir plusieurs MCS possibles

Explicabilité de l’incohérence en PPC

Relation duale entre MUS et MCS

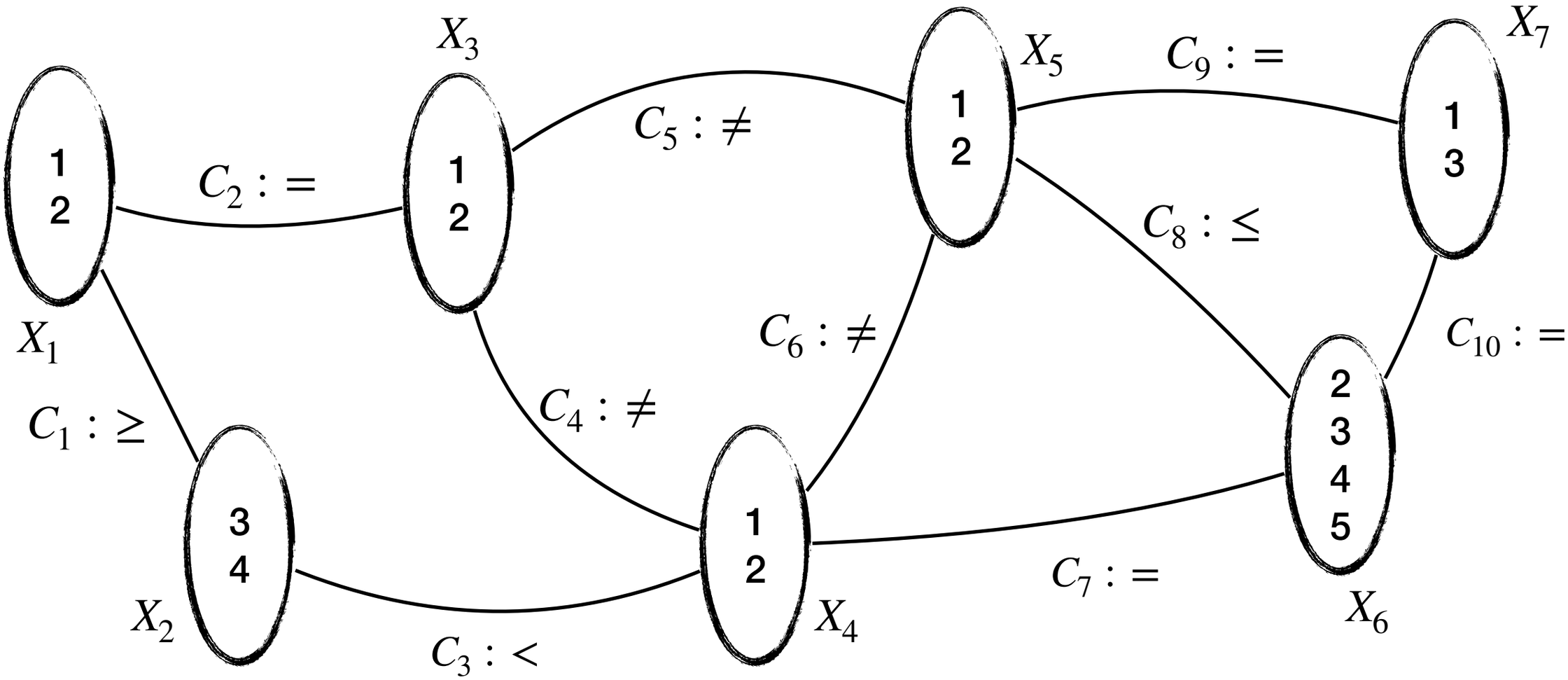
- Chaque MUS est contenu dans au moins un MCS, et chaque MCS intersecte tous les MUS

- **En d’autres termes :**
  - Pour casser une incohérence (MUS), il faut supprimer au moins une contrainte de chaque MUS → ce que fait un MCS
  - Donc, un MCS est un hitting set minimal (ensemble transversal) de l’ensemble des MUS :

MCS $\subseteq C$ tel que $\forall$ MUS M, $M \cap$ MCS $\neq \emptyset$

Explicabilité de l'incohérence en PPC

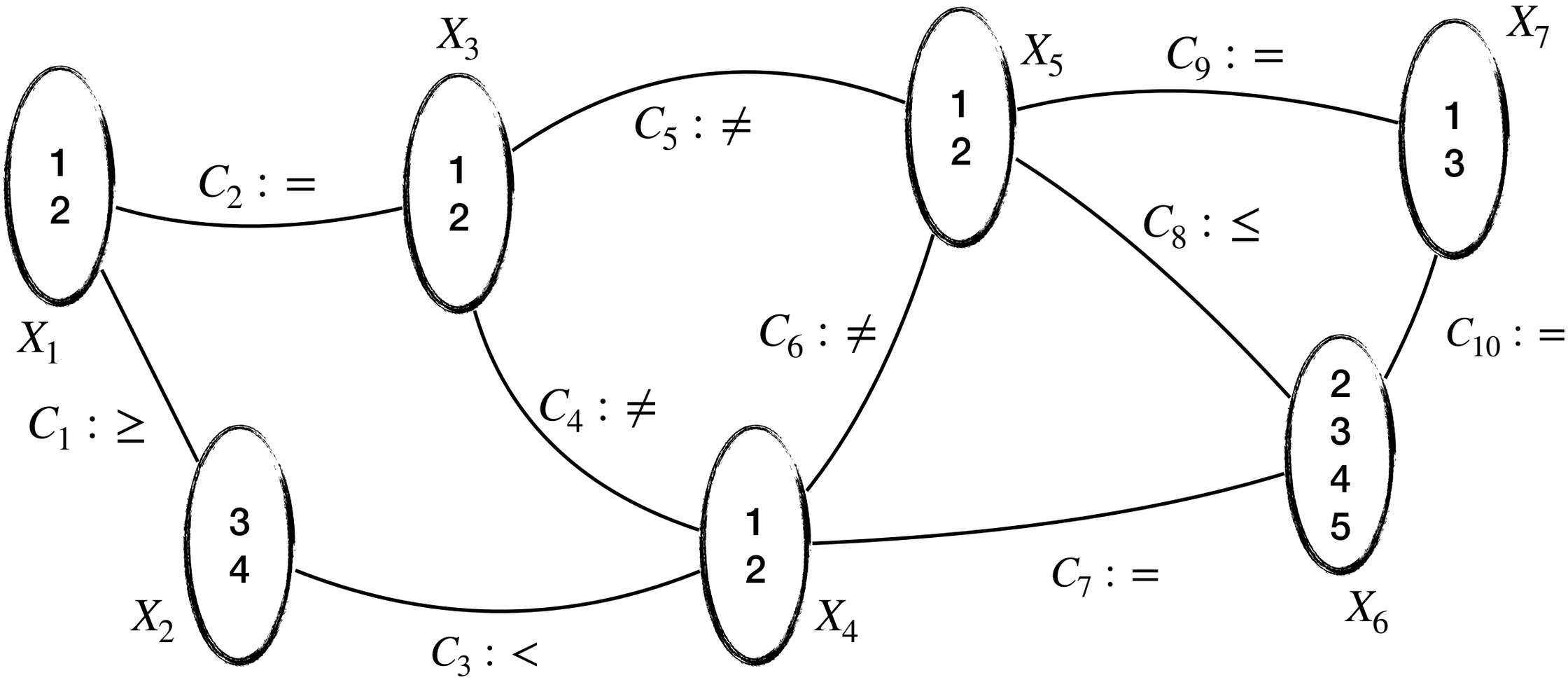
Les Corrections



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections

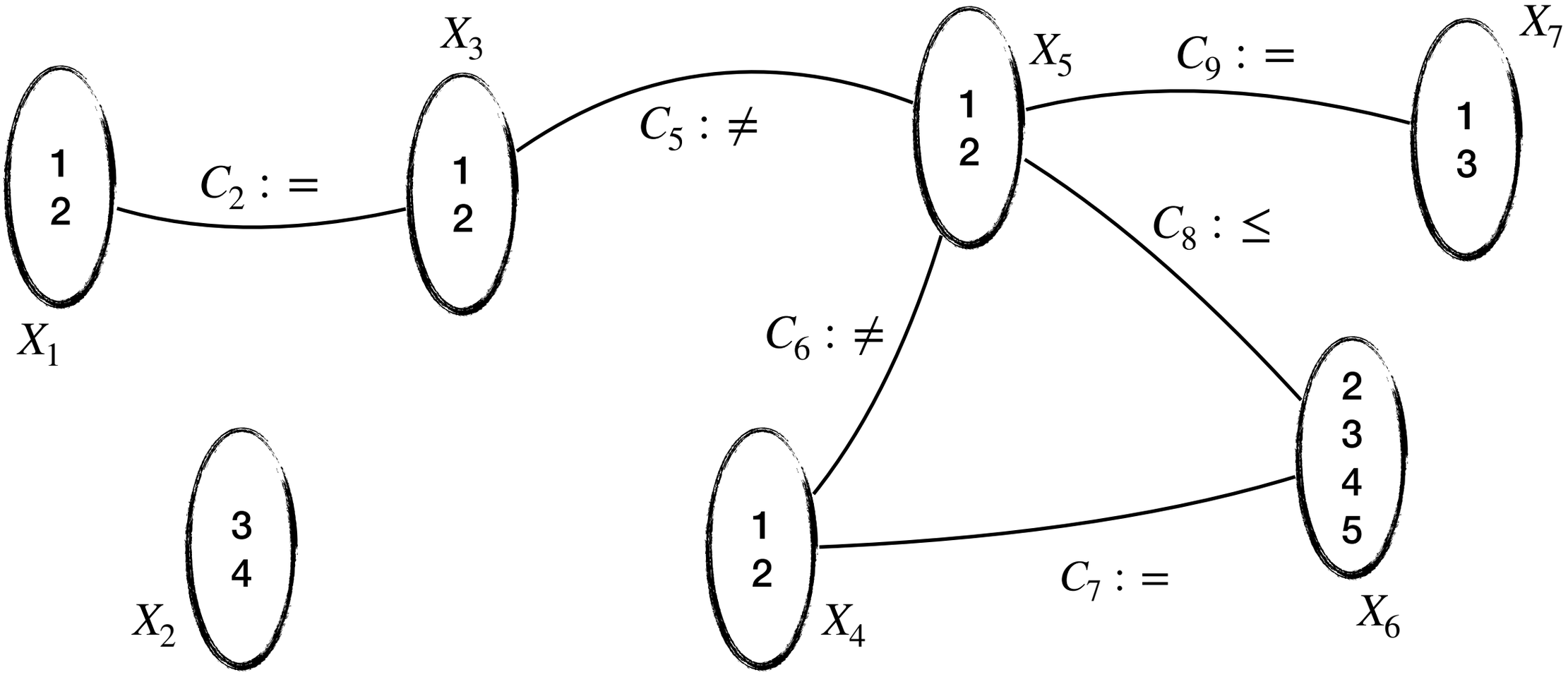


MCS

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections



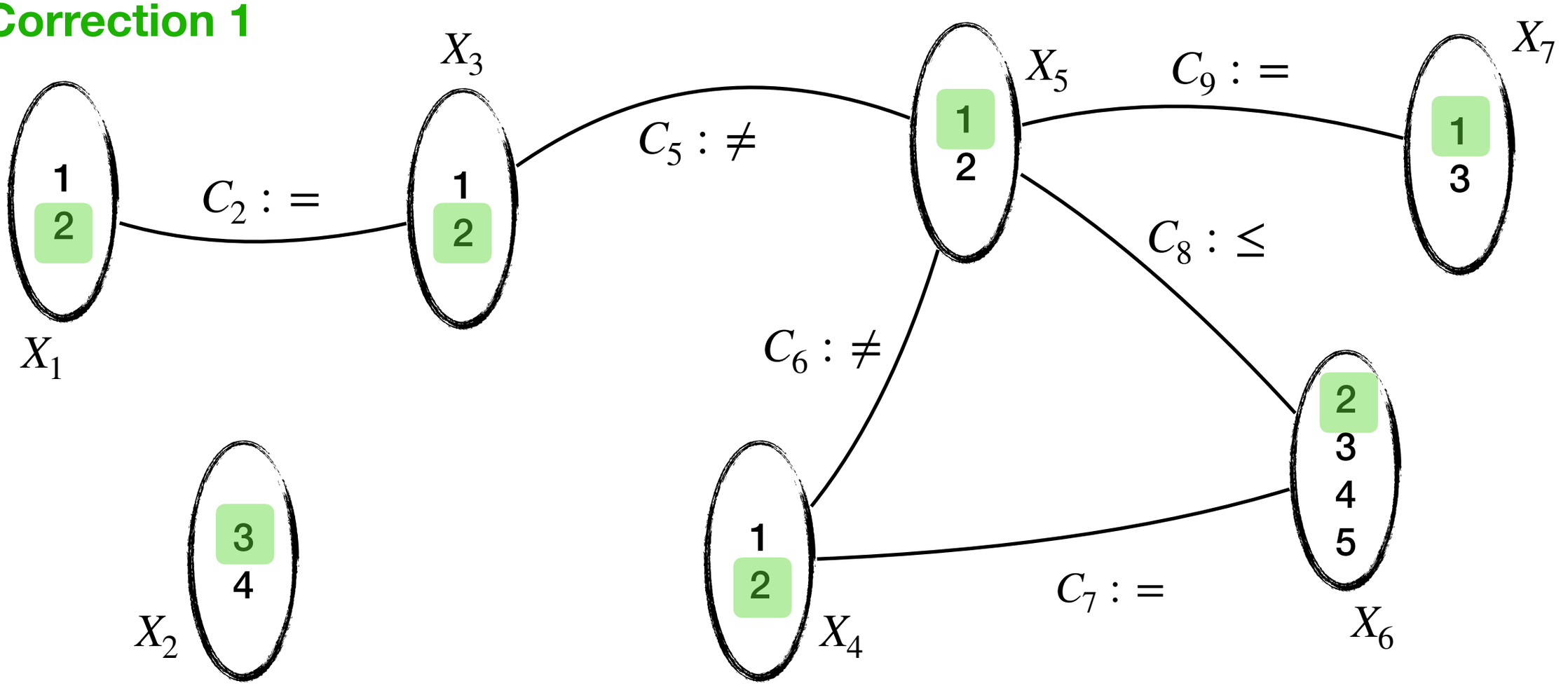
MCS

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections

Correction 1

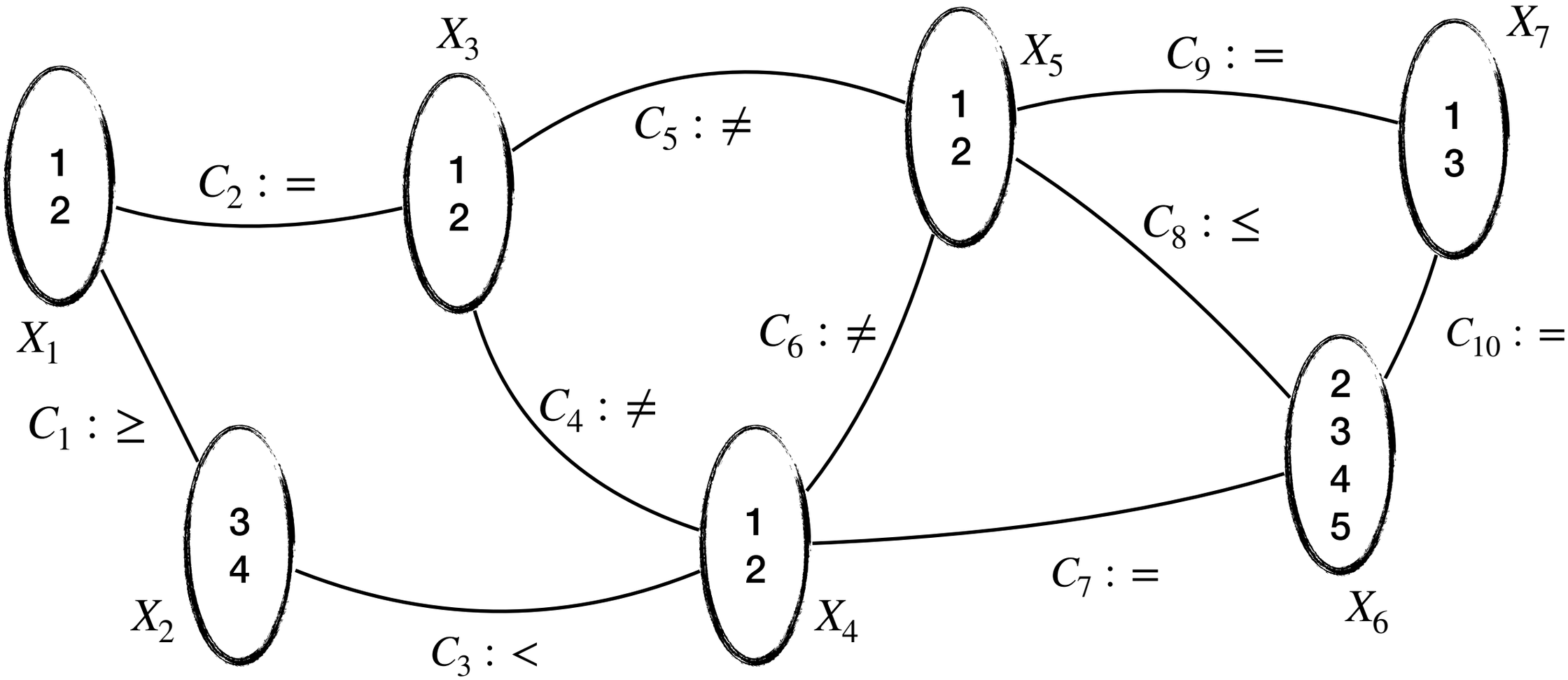


MCS

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections

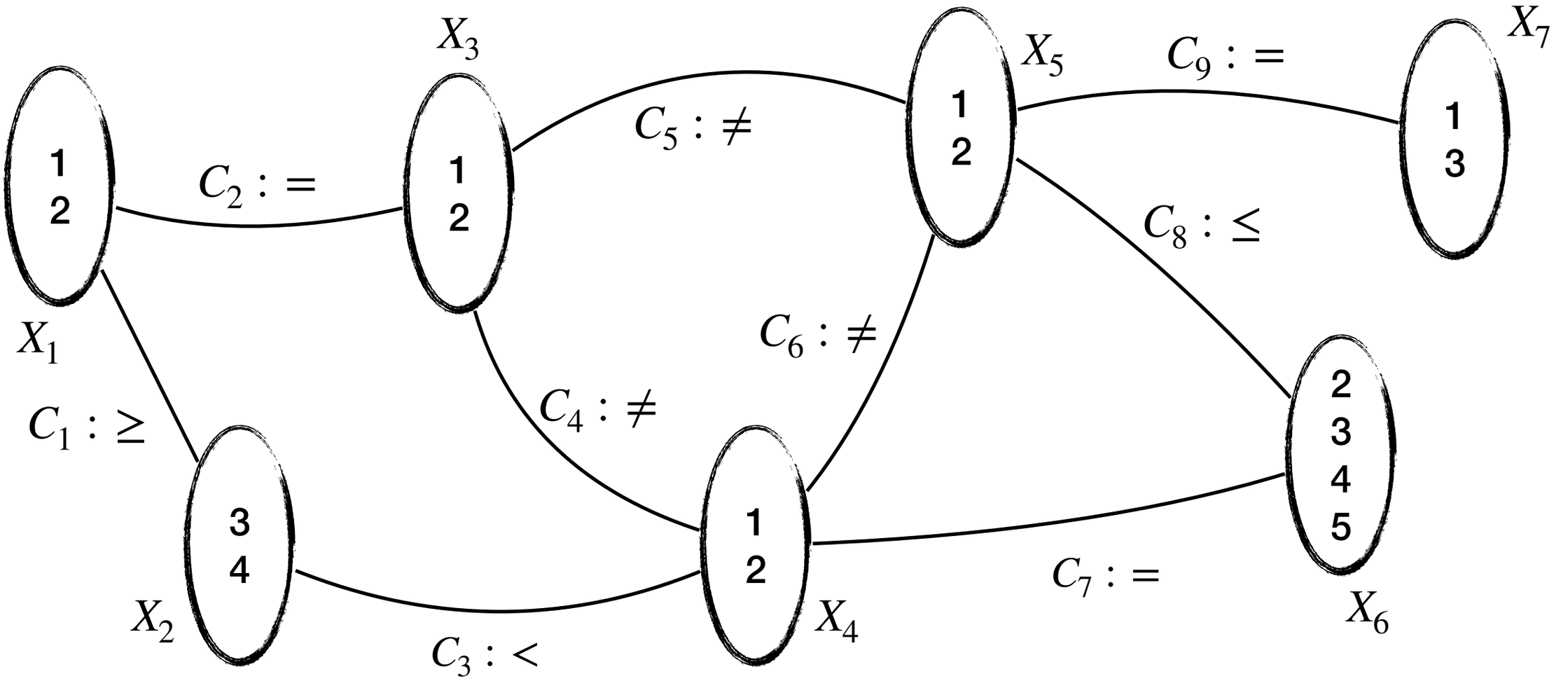


MCS

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections

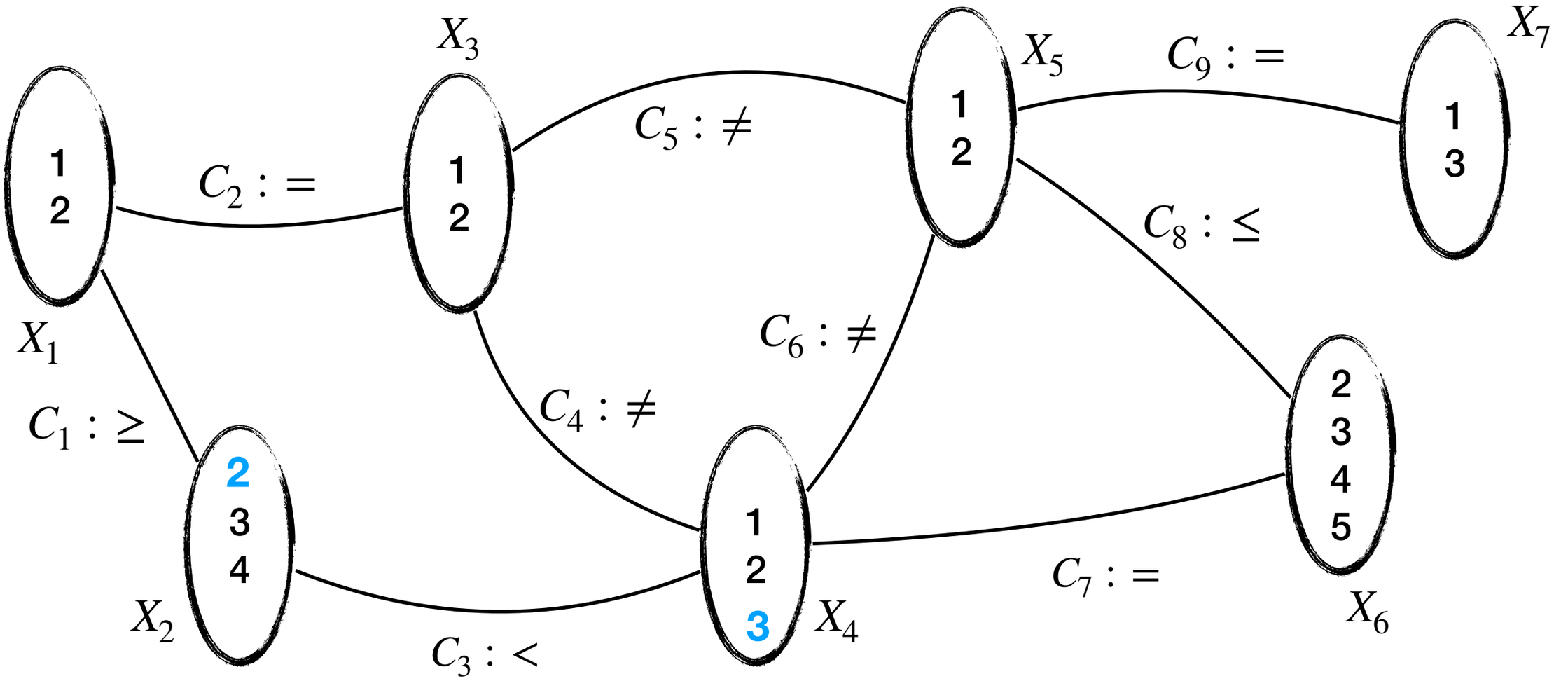


MCS

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections



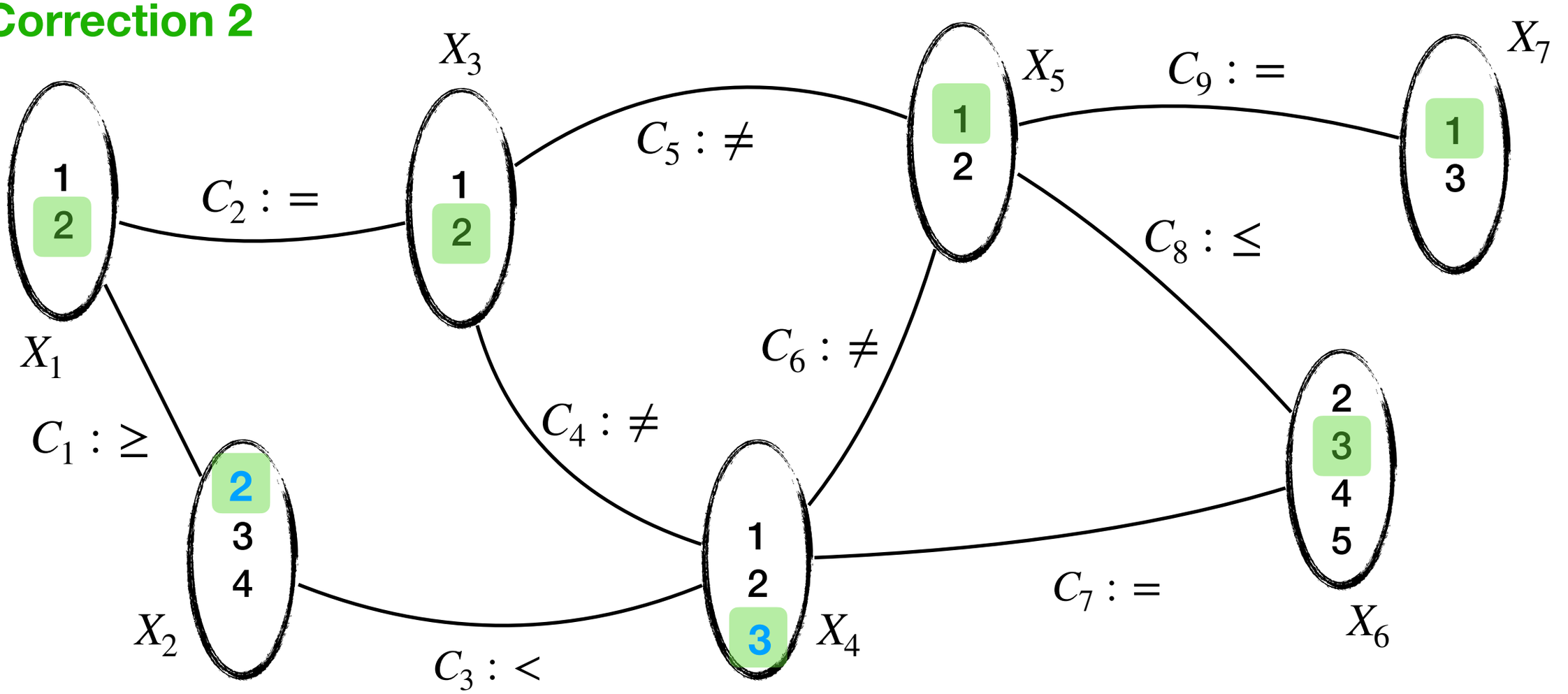
MCS

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections

Correction 2



MCS

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l’incohérence en PPC

Dualité entre MUS et MCS

- Ensemble de tous les **MUS** → génèrent les **MCS** comme hitting sets minimaux

- Ensemble de tous les **MCS** → génèrent les **MUS** comme hitting sets minimaux aussi (dualité parfaite dans certaines classes de problèmes)

- **MUS** : ce qui **cause** l’incohérence

- **MCS** : ce qu’il faut **supprimer** pour restaurer la cohérence.

- **Ces deux notions forment une base fondamentale de l’explicabilité des incohérences dans les modèles à contraintes**

Explicabilité de l'incohérence en PPC

MSS — Maximal Satisfiable Subset

- Soit C un ensemble de contraintes **insatisfiable**. Un sous-ensemble $S \subseteq C$ est un Maximal Satisfiable Subset (MSS) si :

  1. S est **satisfiable**
  2. S est **maximal** pour l'inclusion :

$$\forall c \in C \backslash S, \quad S \cup \{c\} \text{ est insatisfiable}$$

- Autrement dit, on ne peut **ajouter aucune contrainte** à S sans provoquer une incohérence

Explicabilité de l'incohérence en PPC

MSS — Maximal Satisfiable Subset

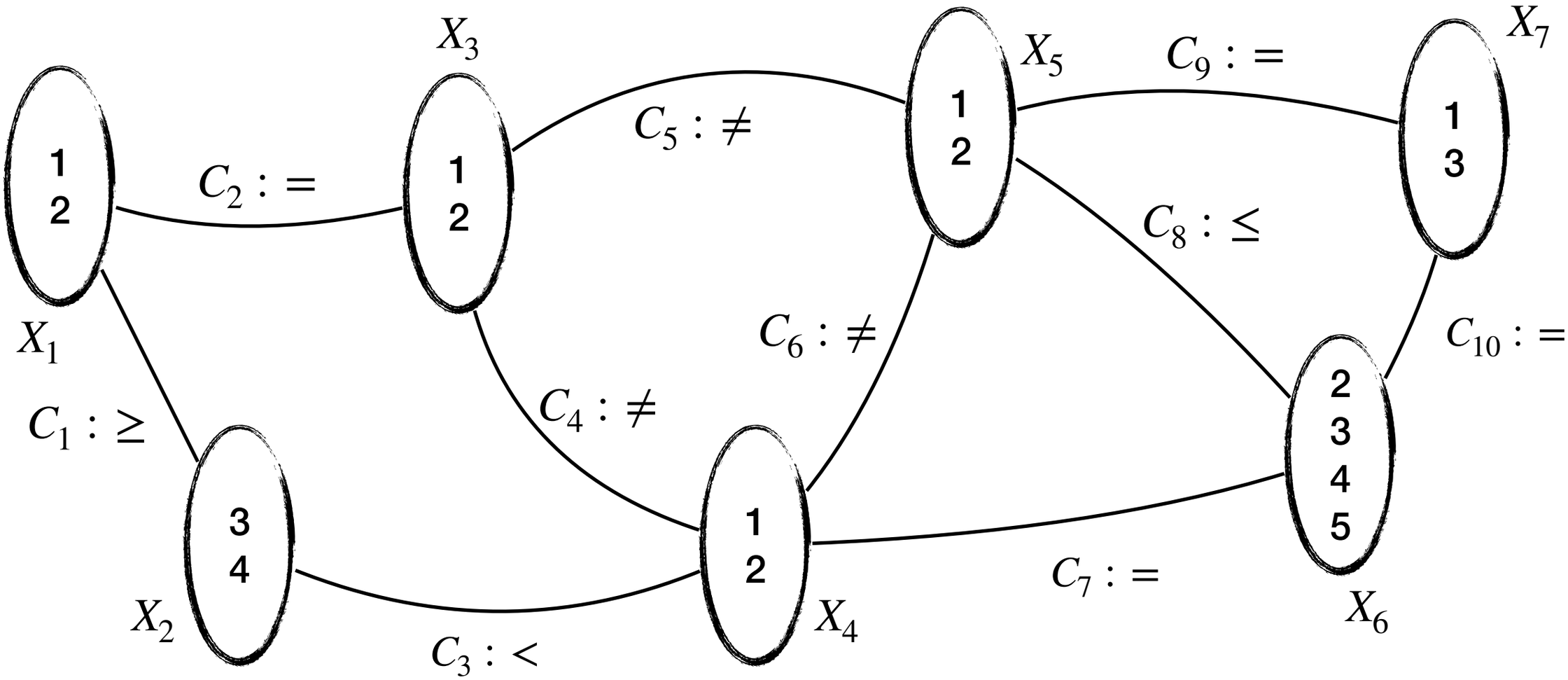
- Un **MSS** est un **plus grand sous-ensemble cohérent** que l'on peut extraire d'un ensemble de contraintes incohérent

- C'est le complément d'un MCS : MSS $= C \setminus$ MCS

- Chaque MSS correspond à une manière cohérente de satisfaire le maximum de contraintes sans tomber dans l'incohérence

Explicabilité de l'incohérence en PPC

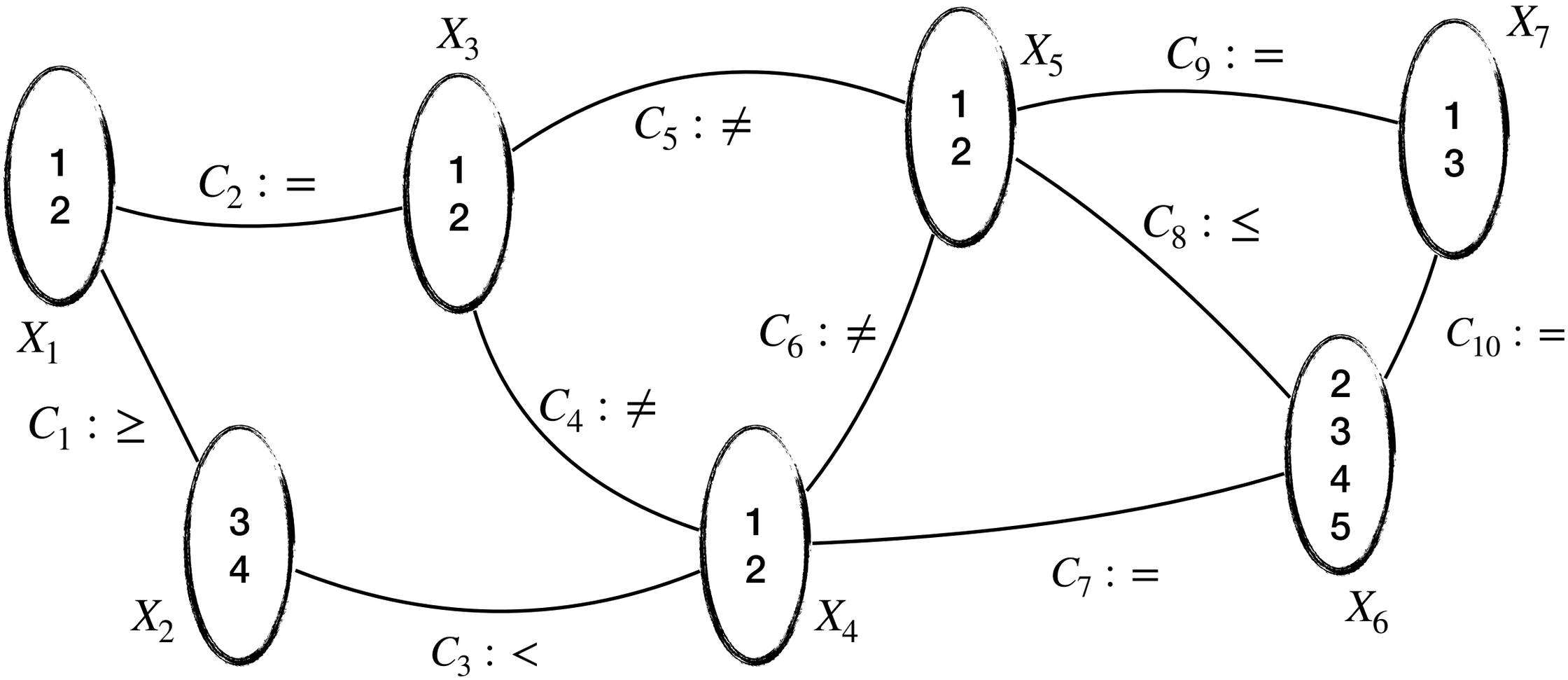
Les Corrections



$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections

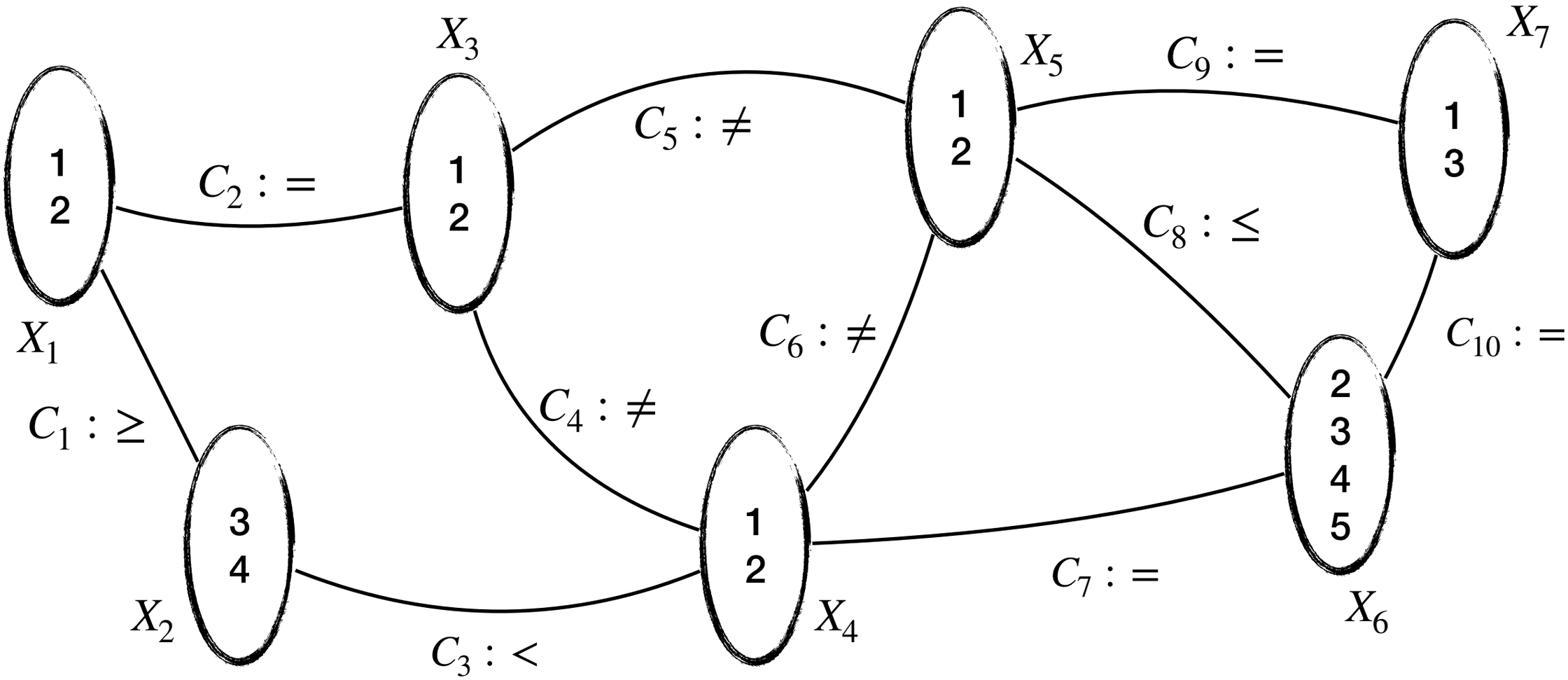


MUSs

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections



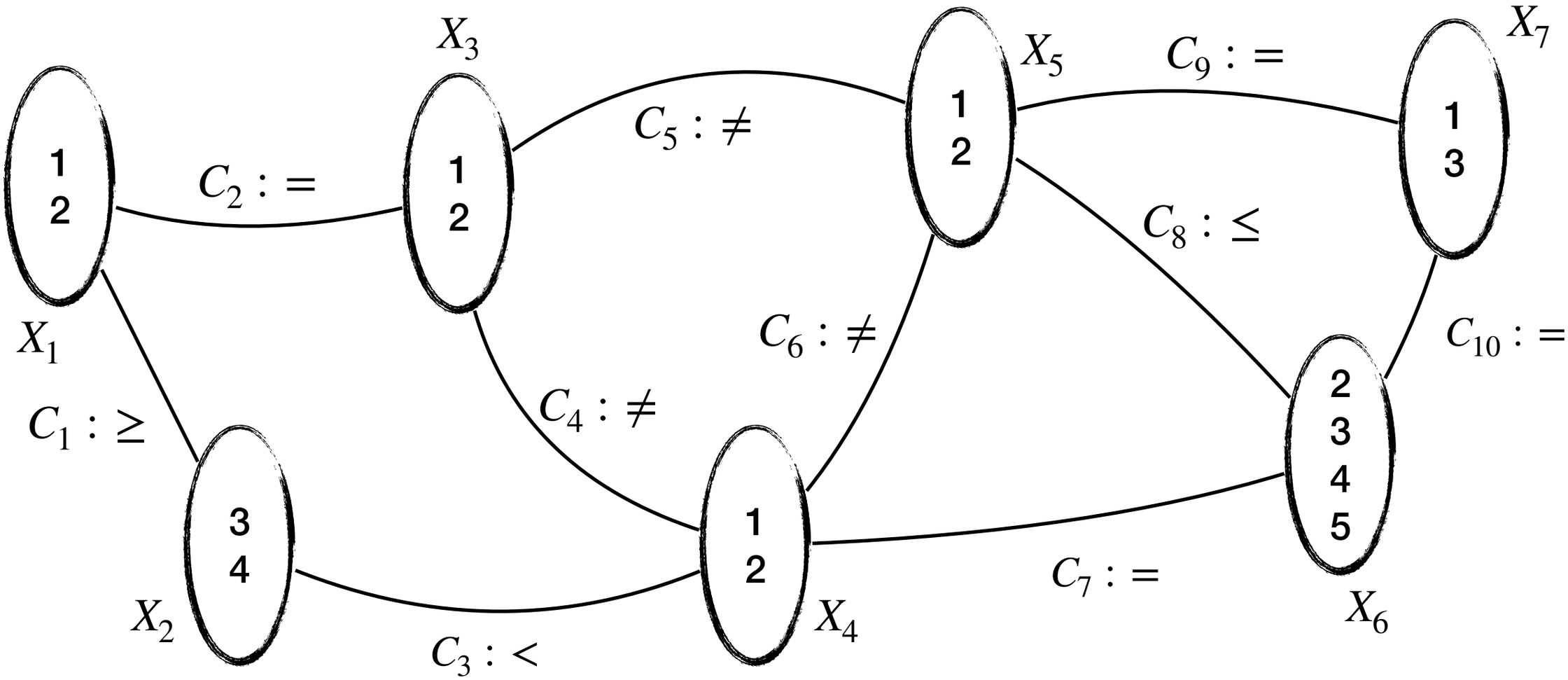
MCS

MUSs

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Les Corrections



MCS **MSS** **MUSs**

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Introduction à QuickXplain

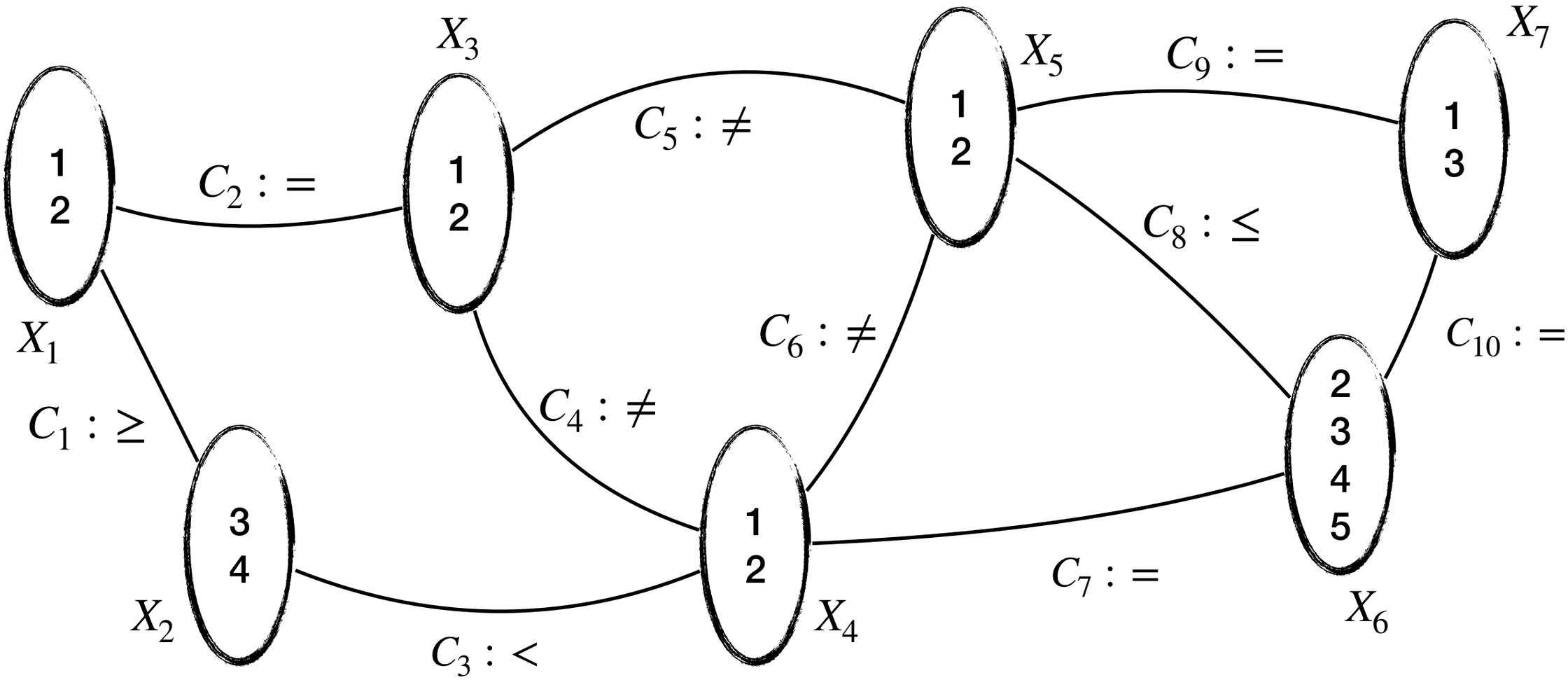
- **QuickXplain** est un algorithme d'explicabilité utilisé pour trouver le plus petit sous-ensemble de contraintes qui rendent un ensemble de contraintes **insatisfiable**

- Cet algorithme se base sur la recherche d'un sous-ensemble minimal **(un MUS)** qui rend l'ensemble **incohérent** lorsqu'il est ajouté à un ensemble de contraintes C donné

- **Utilité : Diagnostic d'incohérence : Identifier les contraintes minimales responsables de l'échec d'un modèle ou d'une planification.**

Explicabilité de l'incohérence en PPC

Les Corrections



MUSs

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Explicabilité de l'incohérence en PPC

Introduction à QuickXplain

Algorithm 1 QuickXplain Algorithm

Input: X, B
Output: X'
if $sol(B) = \emptyset$ **then**
  return $\emptyset$
end if
if $|X| = 1$ **then**
  return X
  $X_1, X_2 \leftarrow \text{split}(X)$
  $X'_1 \leftarrow \text{QuickXplain}(X_1,\ B \cup X_2)$
  $X'_2 \leftarrow \text{QuickXplain}(X_2,\ B \cup X'_1)$
  return $X'_1 \cup X'_2$
end if

Explicabilité de l'incohérence en PPC

Introduction à QuickXplain

Algorithm 1 QuickXplain Algorithm

Input: X (ensemble de contraintes), B (ensemble des contraintes déjà vérifiées)
Output: X' (le sous-ensemble minimal de X rendant $X' \cup B$ inconsistant)
if $sol(B) = \emptyset$ **then**
  return $\emptyset$ {Si B est déjà inconsistant, retourner un ensemble vide}
end if
if $|X| = 1$ **then**
  return X {Si X est indécomposable, retourner X}
else
  $X_1, X_2 \leftarrow \text{split}(X)$ {Diviser X en deux sous-ensembles X_1 et X_2}
  $X'_1 \leftarrow \text{QuickXplain}(X_1, B \cup X_2)$ {Appliquer QuickXplain sur X_1 avec $B \cup X_2$}
  $X'_2 \leftarrow \text{QuickXplain}(X_2, B \cup X'_1)$ {Appliquer QuickXplain sur X_2 avec $B \cup X'_1$}
  return $X'_1 \cup X'_2$ {Retourner l'union des sous-ensembles explicatifs}
end if

Explicabilité de l'incohérence en PPC

Déroulement de QuickXplain (au tableau)

MUSs

$$C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8 \wedge C_9 \wedge C_{10}$$

Algorithm 1 QuickXplain Algorithm

Input: X, B
Output: X'
if $sol(B) = \emptyset$ **then**
  return $\emptyset$
end if
if $|X| = 1$ **then**
  return X
  $X_1, X_2 \leftarrow \text{split}(X)$
  $X'_1 \leftarrow \text{QuickXplain}(X_1,\ B \cup X_2)$
  $X'_2 \leftarrow \text{QuickXplain}(X_2,\ B \cup X'_1)$
  return $X'_1 \cup X'_2$
end if

Artificial Intelligence

Cours8 - Explicabilité en IA

L3 - Informatique

Nadjib Lazaar
Ing - Phd - HDR - Professor - Paris-Saclay University - LISN - LaHDAK
lazaar@lisn.fr https://perso.lisn.upsaclay.fr/lazaar/
14/03/2025