1 Constraint Graphs

Primal Representation of a CSP with a Graph

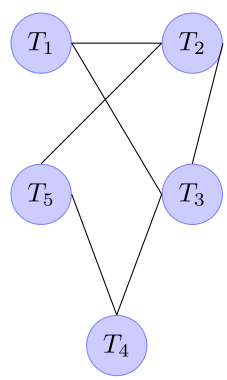
Primal Constraint Graph
Given CSP $P = (V, D, C)$, the **primal constraint graph** of P is the undirected simple graph $G = (V, E)$, where $(x, y) \in E$ iff there is some constraint $c \in C$ for which $x, y \in \text{var}(c)$.

Primal Constraint Graph Example

Defining the Problem

- **Variables:** $T_1, \ldots, T_5$, where T_i is the starting time for a processor to begin work on task i.
- **Constraints:** $T_1 < T_2$, $T_1 < T_3$, $T_2 < T_3$, $T_5 < T_2$, $T_4 < T_5$, $T_4 < T_3$

Primal Constraint Graph



Constraints: $T_1 < T_2$, $T_1 < T_3$, $T_2 < T_3$, $T_5 < T_2$, $T_4 < T_5$, $T_4 < T_3$

Importance of the Primal Graph

The Primal Graph Sheds Light on the CSP

- **Connectivity.** The components of the primal graph represent **independent** problems.
- **Tree Decompositions.** Removing some edges from the primal graph to form a tree can result in a polynomial-time solvable sub-problem.
- **Isomorphism.** Isomorphic primal graphs suggest similar types of problems. Solving one problem might shed light on solving the other problem.
- **Vertex degrees** help identify efficient ordering of the variables when solving the problem via backtracking.

Dual Representation of a CSP with a Graph

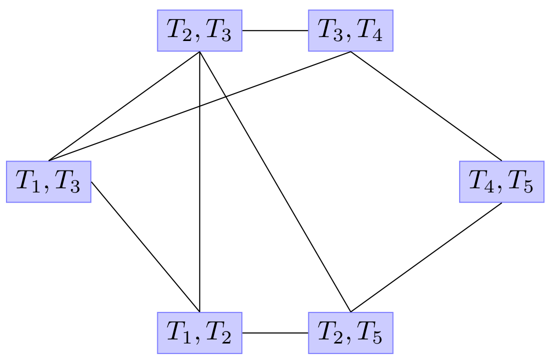
Dual Constraint Graph
Given CSP $P = (V, D, C)$, the **dual constraint graph** of P is the undirected simple graph whose vertex set is the set of all distinct constraint scopes of the constraints in C, and for which two scopes are adjacent iff they have one or more variables in common.

Dual Constraint Graph Example

Defining the Problem

- **Variables:** $T_1, \ldots, T_5$, where T_i is the starting time for a processor to begin work on task i.
- **Constraints:** $T_1 < T_2$, $T_1 < T_3$, $T_2 < T_3$, $T_5 < T_2$, $T_4 < T_5$, $T_4 < T_3$

Dual Constraint Graph



Constraints: $T_1 < T_2$, $T_1 < T_3$, $T_2 < T_3$, $T_5 < T_2$, $T_4 < T_5$, $T_4 < T_3$

2 The Dual of a Constraint Problem

The Dual of a Constraint Problem

Dual Constraint Graph
Given CSP $P = (V, D, C)$, the **dual of** P, denoted $P^\perp$ is the constraint problem with the following properties.

- **Variables:** $c \in C$, for each constraint c of P.
- **Domains:** $\text{dom}(c) = c$ (remember, c is a subset of assignments!).
- **Constraints:** for each pair $c_i, c_j \in C$ for which $\text{var}(c_i) \cap \text{var}(c_j) \neq \emptyset$, (a_1, a_2) satisfies the dual constraint c_{ij} iff $a_1 \in c_i$, $a_2 \in c_j$, and a_i and a_j agree at $\text{var}(c_i) \cap \text{var}(c_j)$.

Dual Constraint Problem Example

- $c_1 = \{(1,2,1),(1,3,2),(2,1,4),(1,4,2)\}$ is a constraint of P over $\{x,y,z\}$.
- $c_2 = \{(1,2,3),(2,4,1),(1,2,2),(1,4,2)\}$ is a constraint of P over $\{x,z,w\}$.
- c_{12} is a constraint over variables c_1 and c_2 of $P^{\perp}$.
- Satisfying assignments of c_{12}:

$$\{((1,3,2),(1,2,3)),((1,3,2),(1,2,2)),((1,4,2),(1,2,3)),$$
$$((1,4,2),(1,2,2)),((2,1,4),(2,4,1))\}$$

Easier way to Extensionally Represent a Dual Constraint

Use the assignments from $c_1 \bowtie c_2$

- $c_1 = \{(1,2,1),(1,3,2),(2,1,4),(1,4,2)\}$ is a constraint of P over $\{x,y,z\}$.
- $c_2 = \{(1,2,3),(2,4,1),(1,2,2),(1,4,2)\}$ is a constraint of P over $\{x,z,w\}$.
- Satisfying assignments of c_{12}:

$$\{((1,3,2),(1,2,3)),((1,3,2),(1,2,2)),((1,4,2),(1,2,3)),$$
$$((1,4,2),(1,2,2)),((2,1,4),(2,4,1))\}$$

- Written as join tuples: $\{(1,3,2,3),(1,3,2,2),(2,1,4,1),(1,4,2,3),(1,4,2,2)$

3 Binary Constraint Networks

Binary Constraint Networks

- A **binary constraint network** is a constraint satisfaction problem for which each constraint is binary (i.e. is a relation over exactly two variables).
- Not every problem can be represented by a binary constraint network.
- P_1 and P_2 are **logically equivalent** iff $\text{sol}(P_1) = \text{sol}(P_2)$.

Counting Arguments

- Assume n variables, each with a domain of size k.
- Number of possible constraint problems (up to logical equivalence): 2^{k^n}
- Number of binary constraint networks: $(2^{k^2})^{n^2} = 2^{k^2 n^2}$

Projection Networks

- Given relation R over V, then **projection network** of R, denoted $\pi(R)$, is the binary constraint network over variable set V, and whose constraints are $c_{xy} = \pi_{\{x,y\}}(R)$.
- In other words, the constraints of $\pi(R)$ are obtained by projecting the assignments of R onto each pair of variables of V.

Projection Network Example

- $R = \{(2,3,2),(2,2,3),(2,3,3)\}$ is a relation over $\{x,y,z\}$.
- Constraints of $\pi(R)$: $c_{xy} = \{(2,2),(2,3)\}$, $c_{xz} = \{(2,2),(2,3)\}$, $c_{yz} = \{(2,3),(3,2),(3,3)\}$
- For this Example: $\text{sol}(\pi(R)) = R$

Projection Network Example 2

- $R = \{(1,1,1),(1,2,3),(2,1,1),(2,2,1)\}$ is a relation over $\{x,y,z\}$.
- Constraints of $\pi(R)$: $c_{xy} = \{(1,1),(1,2),(2,1),(2,2)\}$, $c_{xz} = \{(1,1),(1,3),(2,1)\}$, $c_{yz} = \{(1,1),(2,1),(2,3)\}$
- $\text{sol}(\pi(R)) = \{(1,1,1),(1,2,1),(1,2,3),(2,1,1),(2,2,1)\}$ has more assignments than R.

Some Facts About Projection Networks

- For all relations R, $R \subseteq \text{sol}(\pi(R))$.
- Any binary constraint network P for which $R \subseteq \text{sol}(P)$ must contain $\pi(R)$.
- Hence, $\pi(R)$ is the minimum binary constraint network whose set of solutions contains R.
- Therefore, R can be represented by a binary constraint network iff $R = \text{sol}(\pi(R))$.

Exercises

1. Recall the Graph Coloring Problem. Let $G = (V, E)$ be a simple graph where $V = \{a, b, c, d, e, f, g\}$ and

$$E = \{(a, b), (a, d), (b, c), (b, d), (b, g), (c, g), (d, e), (d, f), (d, g), (f, g)\}.$$

   The problem is to find a coloring of V using colors **red**,**blue**, and **yellow**, so that no two adjacent vertices are assigned the same color. Define a CSP for this problem. Clearly define the variables, domains, and constraints. Find at least one solution to the CSP. Draw the primal graph for this CSP. What do you notice?

2. Repeat Problem 1, but now draw the dual graph of the CSP.

3. Let $c_1 = \{(a, b), (a, c), (b, a), (b, c), (c, a), (c, b)$ be a constraint over $\{x, y\}$, and $c_2 = \{(a, b), (a, c), (b, a), (b, c), (c, a), (c, b)$ a constraint over $\{y, z\}$ for some constraint problem P. List the tuples of the constraint that exists between c_1 and c_2 in $P^{\perp}$.

4. A magic square of order n is an $n \times n$ matrix of the integers $1, 2 \ldots, n^2$, arranged so that the sum of every row, column, and main diagonals add to the same value. Provide a formula for the value that the rows, columns, and diagonals should add to. Prove your answer. Define a CSP P for the case $n = 3$. Draw the primal and dual graphs of P. Either solve by hand, or use MiniZinc to decide if this problem (for $n = 3$) has a solution.

5. Consider the following relation over $\{x, y, z, t\}$:

$$R = \{(a, a, a, a), (a, b, b, b), (b, b, a, c)\}.$$

   (a) Compute $\pi(R)$.

   (b) Is R representable by a binary consraint network? Justify your answer.

   (c) Same question for the relation $\pi_{xyz}(R)$.

   (d) A **partial solution** to a CSP $P = (V, D, C)$ is any assignment a over a subset of V that is consistent with every $c \in C$. Provide a partial solution$a$ for $\pi(R)$ that is defined over $\{x, y\}$. Verify that a is consistent with all constraints.

   (e) We say that a CSP has a **backtrack-free** ordering $\{x_1, \ldots, x_n\}$ of its variables iff any partial solution defined over $\{x_1, \ldots, x_i\}$ can be extended to a partial solution over $\{x_1, \ldots, x_i, x_{i+1}\}$. Find a backtrack-free ordering of $\{x, y, z, t\}$ for $\pi(R)$.

   (f) Show by counterexample that $\{x, y, z, t\}$ itself is *not* a backtrack-free ordering for $\pi(R)$.