Constraint Satisfaction Problems: Introduction

Alice Gao

Lecture 5

Based on work by K. Leyton-Brown, K. Larson, and P. van Beek

Outline

Learning Goals

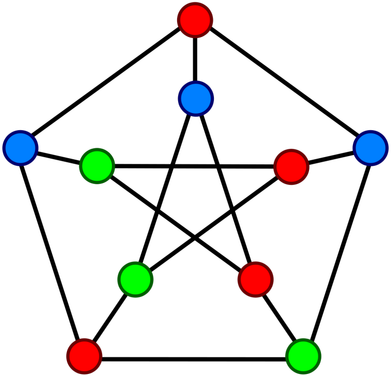
By the end of the lecture, you should be able to

- Describe components of a constraint satisfaction problem.
- Formulate a real-world problem as a constraint satisfaction problem.
- Formulate a constraint using the table or the formula format.
- Verify whether a variable is arc-consistent with respect to another variable for a constraint.
- Define/implement/trace the arc consistency algorithm. Describe the possible outcomes of the arc-consistency algorithm.
- Analyze the complexity of the arc consisteny algorithm.

Example: Crossword Puzzles

T	A	O	S	■	S	P	E	C	S	■	S	E	P	T
E	G	G	Y	■	A	L	A	A	P	■	I	L	I	A
T	H	R	E	E	P	E	N	N	Y	■	D	H	A	K
H	A	E	■	X	E	B	E	C	■	S	E	I	N	E
■	■	■	V	O	L	■	D	H	O	W	S	■	■	■
S	W	E	E	N	E	Y	■	A	N	A	T	M	A	N
P	R	Y	S	■	S	O	N	■	E	P	E	I	R	A
L	Y	R	I	C	■	N	O	S	■	S	P	A	E	R
I	L	I	C	E	S	■	R	U	C	■	P	O	C	K
T	Y	R	A	N	E	D	■	R	O	P	E	W	A	Y
■	■	■	T	S	A	R	S	■	C	U	D	■	■	■
B	A	G	I	E	■	O	I	N	K	S	■	T	S	K
O	L	E	O	■	S	W	E	E	P	S	T	A	K	E
Y	A	R	N	■	E	N	V	O	I	■	A	B	U	T
G	Y	M	S	■	I	D	E	N	T	■	G	U	G	A

Example: Graph Coloring Problem



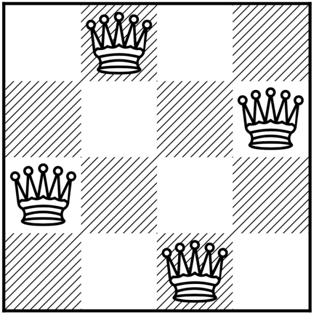
Applications:

- Designing seating plans
- Exam scheduling
- ...

Example: Sudoku

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Example: 4-Queens Problem



Introduction to CSPs

- So far, we solve problems by searching in a state space.
- The algorithm is unaware of the structure of the states.
- Can we develop efficient general-purpose algorithms, which take advantage of the structure of states?

Definition of a CSP

- A set X of variables: $\{X_1, X_2, ..., X_n\}$
- A set D of domains: D_i is the domain for variable X_i, $\forall i$.
- A set C of constraints specifying allowable combinations of values

A solution is an assignment of values to all the variables that satisfy all the constraints.

Defining Constraints

Constraints restrict the values that one or more variables can take.

- The arity of a constraint is the number of variables involved in a constraint.
- An unary constraint involves one variable.
- A k-ary constraint involves k variables.

Variants of a CSP

We may want to solve the following problems with a CSP:

- Determine whether a solution exists or not.
- Find one solution.
- Find all the solutions.
- Find the optimal solution, given some cost function.

Example: Sudoku

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

CQ: Rewriting row constraints

CQ: Consider the CSP formulation for the Sudoku problem.
Is it possible to rewrite each row constraint into

(A) A set of unary constraints

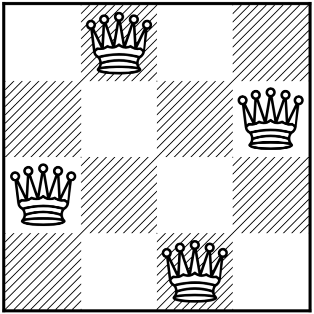
(B) A set of binary constraints

(C) A set of tertiary (3-ary) constraints

(D) Two of (A), (B), and (C)

(E) All of (A), (B), and (C)

Example: 4-Queens Problem



CQ: Constraints for 4-Queens Problem

CQ: Given the definitions of variables and their domains for the 4-queens problems, which constraints do we need to define?

(A) No two queens can be in the same row.

(B) No two queens can be in the same column.

(C) No two queens can be in the same diagonal.

(D) Two of (A), (B), and (C)

(E) All of (A), (B), and (C)

Defining Constraints

There are two ways of defining a constraint.

- The list/table format:
  Give a list/table of values of the variables that satisfy the constraints.
- The function/formula format:
  Give a function/formula, which returns/is true if the values of the variables satisfy the constraint.

CQ: Defining Constraints as a Table

CQ: Suppose that we use a 2-column table to encode the following constraint. In each row of the table, the two values of x_0 and x_2 satisfy the constraint.

> *The two queens in columns 0 and 2 are not in the same row or diagonal.*

How many rows are there in this table?

(A) Less than 8

(B) 8

(C) 9

(D) 10

(E) More than 10

CQ: Defining Constraints as a Formula

CQ: Suppose that we encode the following constraint as a propositional formula.

> *The two queens in columns 0 and 2 are not in the same row or diagonal.*

Which of the following formula is correct?

(A) $(x_0 \neq x_2)$

(B) $((x_0 \neq x_2) \wedge ((x_0 - x_2) \neq 1))$

(C) $((x_0 \neq x_2) \wedge ((x_0 - x_2) \neq 2))$

(D) $((x_0 \neq x_2) \wedge (|x_0 - x_2| \neq 1))$

(E) $((x_0 \neq x_2) \wedge (|x_0 - x_2| \neq 2))$

Solving a CSP

When solving a CSP, we can combine

- Search, and
- Inference, called Constraint Propagation

Constraint Propagation

Eliminate the values of a variable that are inconsistent with the constraints involving the variable.

Consistency for Different Constraints

- Unary constraints:
- Binary constraints:
- k-ary constraints where $k \geq 2$:

Definition of Arc Consistency

Definition (Arc Consistency)

The variable X_i is arc-consistent with respect to another variable X_j if and only if for every value v_i in D_i, there is a value v_j in D_j such that (v_i, v_j) satisfies the constraint (X_i, X_j).

If X_i is not arc-consistent with the variable X_j, we can make it consistent by removing values in D_i that is not consistent with any value on D_j. This removal can never rule out any solution.

CQ: Definition of Arc Consistency

CQ: Consider the constraint "X is divisible by Y" between two variables X and Y. X is arc-consistency with respect to Y in how many of the four scenarios below?

1. $dom(X) = \{10, 12\}, dom(Y) = \{3, 5\}$
2. $dom(X) = \{10, 12\}, dom(Y) = \{2\}$
3. $dom(X) = \{10, 12\}, dom(Y) = \{3\}$
4. $dom(X) = \{10, 12\}, dom(Y) = \{3, 5, 8\}$

(A) 0 (B) 1 (C) 2 (D) 3 (E) 4

CQ: Definition of Arc Consistency

CQ: True or False:

If X is arc-consistent with respect to Y,
then Y is arc-consistent with respect to X.

(A) True

(B) False

(C) Not enough information to tell

CQ: Definition of Arc Consistency

CQ: Assume that X is arc-consistent with respect to Y.
Remove one value from the domain of Y.
Is X still arc-consistent with respect to Y?

(A) Yes

(B) No

(C) Not enough information to tell

CQ: Definition of Arc Consistency

CQ: Assume that Y is arc-consistent with respect to Z.
Remove one value from the domain of Y.
Is Y still arc-consistent with respect to Z?

(A) Yes

(B) No

(C) Not enough information to tell

Making (X_i, C) arc-consistent

Let C be a constraint between the variables X_i and X_j.

Algorithm 1 Revise(X_i, C)

revised $\leftarrow$ false
for x in $dom(X_i)$ **do**
    if $\neg\exists\, y \in dom(X_j)$ s.t. (x, y) satisfies the constraint C **then**
        remove x from $dom(X_i)$
        revised $\leftarrow$ true
    end if
end for
return revised

The AC-3 Arc Consistency Algorithm

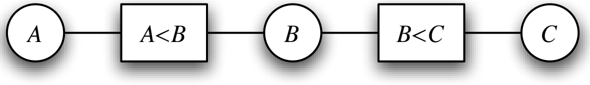
Algorithm 2 The AC-3 Algorithm

Put (v, C) in the set S for every variable v and every constraint involving v.
while S is not empty **do**
    remove (X_i, C_{ij}) from S (C_{ij} is a constraint between X_i and X_j.)
    if Revise(X_i, C_{ij}) **then**
        if $dom(X_i)$ is empty **then return** false
        for X_k where C_{ki} is a constraint between X_k and X_i **do**
            add (X_k, C_{ki}) to S
        end for
    end if
end while
return true

Properties of the Arc Consistency Algorithm

- Does the order in which arcs are considered matter?
- Three possible outcomes of the arc consistency algorithm:
- Time complexity:
  n variables, c binary constraints, and the size of each domain is at most d.

Example: Arc Consistency



$$dom(A) = \{1, 2, 3, 4\}; dom(B) = \{1, 2, 3, 4\}; dom(C) = \{1, 2, 3, 4\}$$

CQ: Number of Items in the Queue

CQ: The start of the arc consistency algorithm says that "put all the arcs in the queue."
After this step, how many items are there in the queue?

(A) 1

(B) 2

(C) 3

(D) 4

(E) Larger than 4

CQ: Adding an Arc into the Queue

CQ: When we remove $(A, A < B)$ and reduce A's domain, should we add $(B, A < B)$ back into the queue?

(A) Yes, always.

(B) No, never.

(C) Yes, if $(B, A < B)$ is not in the queue.

Revisiting the Learning Goals

By the end of the lecture, you should be able to

- Describe components of a constraint satisfaction problem.
- Formulate a real-world problem as a constraint satisfaction problem.
- Formulate a constraint using the table or the formula format.
- Verify whether a variable is arc-consistent with respect to another variable for a constraint.
- Define/implement/trace the arc consistency algorithm. Describe the possible outcomes of the arc-consistency algorithm.
- Analyze the complexity of the arc consisteny algorithm.