CC484 - Constraint Satisfaction

Edgar Galván López and Edward Tsang

University of Essex
Deparment of Computer Science
Wivenhoe Park, Colchester,CO4 3SQ
United Kingdom
egalva@essex.ac.uk

25 October, 2006

Outline

Overview (1/2)

- We regularly encounter constraint in our day-to-day lives.
- The tasks proposed in the language of constraints are computationally intractable (NP-hard).
- There are two components to every constraint problem:
  1. Every constraint problem must include **variables**. The set of possible values for a given variable is called its domain.
  2. **Constraints** are rules to impose a limitation on the values that a variable may be assigned.

Overview (2/2)

- A model that includes variables, their domains and constraints is called a **constraint network** or **constraint problem**.
- What is a solution? Anyway!
- A problem that has one or more solutions is **consistent**.
- If there is no solution, then the network is **inconsistent**.

The problem

- The n-Queens problem is to place n queens on a n by n chessboard.
- The restriction is that no queen can take any other queens.

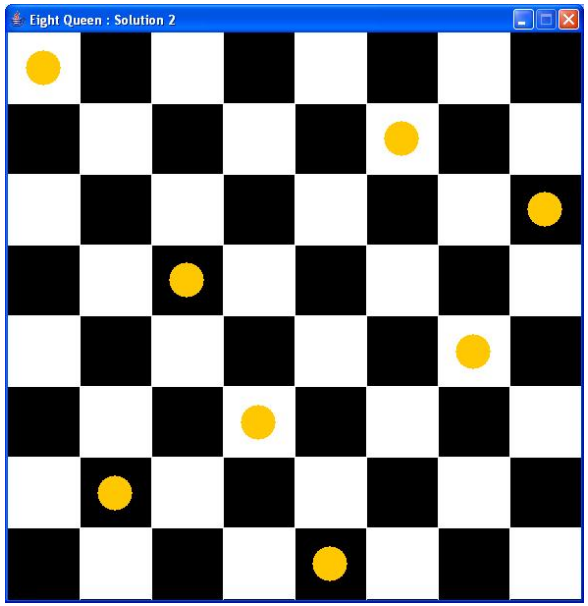

Figure: 1. A solution for the 8-Queens problem.

The Problem

- Possible words: {PATH, LASER, CONSOLATION, CONSISTENT, CROSS, IRON, EXAMINATION, WORD, FLOAT, PALPITATION, TEN, YES, NO, REVEREND, MANIFESTATION}

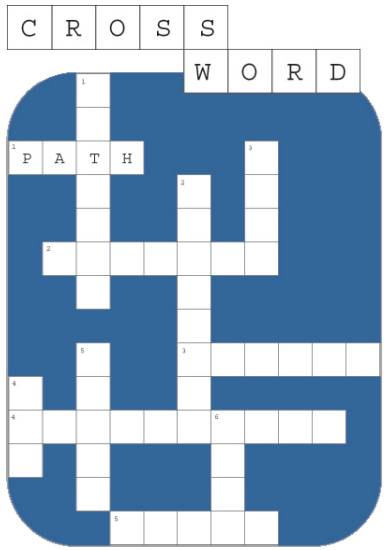

Figure: 2. A crossword puzzle.

The Problem (1/2)

- Five developments are to be located on the lots shown in Figure 3: a recreation area, an apartment complex, a cluster of 50 single-family houses, a large cemetery and a dump site.

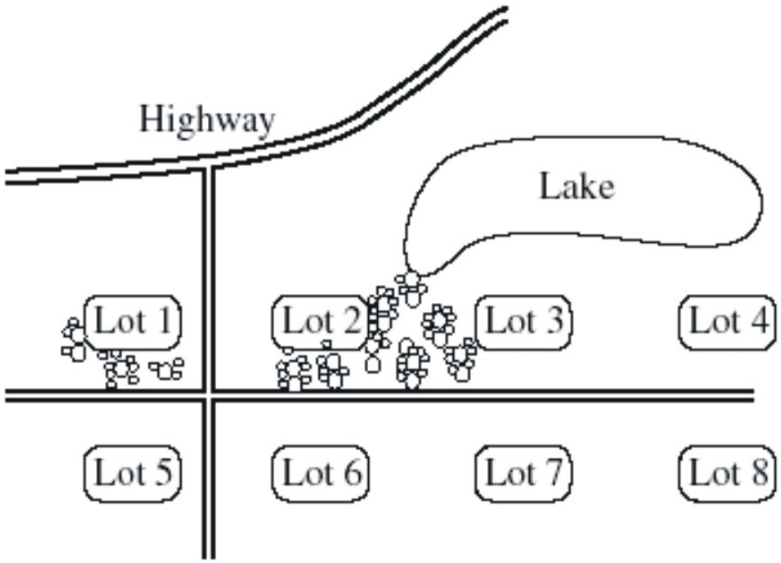

Figure: 3. Map.

The Problem (2/2)

- The recreation area must be near the lake.
- Steep stones must be avoided for all but the recreation area.
- Poor soil must be avoided for developments that involve construction, namely, the apartments and the houses.
- Because it is noisy, the highway must not be near the apartments, the houses or the recreation area.
- The dump site must not be visible from the apartments, the houses, or the lake.
- Lots 3 and 4 have poor soil.
- more...

Definitions (1/3)

- A *set* is a collection of distinguishable objects and a single object is called *member* or *element*.
- If an object x is member of a set A, we write $x \in A$.
- If an object x is not member of a set A, we write $x \notin A$.
- $A = \{1, 2, 3\}$ can be expresed as:
  $A = \{x | x \text{ an integer and } 1 \leq x \leq 3\}$.
- A is a subset of B if each element of set A is also an element of set B ($A \subseteq B$).

Definitions (2/3)

- The *intersection* of two sets A and B is the set $A \cap B = \{x | x \in A \text{ and } x \in B\}$.
- The *union* of two sets A and B is the set $A \cup B = \{x | x \in A \text{ or } x \in B\}$.
- The *difference* of two sets A and B is the set $A - B = \{x | x \in A \text{ and } x \notin B\}$.
- A set contaning no elements is called *empty set*.
- Taking into account that a variable is a collection of values, called a domain, we define *k-tuple* as a sequence of k not necessarily distict objects denoted by $(a_1, \cdots, a_k)$, and a object in the sequence is called component.
- Refer to your exercises (number 1).

Definitions (3/3)

- The Cartesian product or product of a list of domains $D_1, \cdots, D_k$, written $D_1 x \cdots x D_k$, is the set of all $k-tuples$ $a_1, \cdots, a_k$ such that a_1 is in D_1, a_2 is in D_2 and so on.
- Refer to your exercises (number 2(a)).

Definitions (1/1)

- A relation R on the set of variables is any subset of the Cartesian product of their domains.
- The set of variables on which a relation is defined is called the scope of the relation, denoted $scope(R)$.
- Refer to your exercises (number 2(b)).

Definitions (1/1)

- Given two relations, R and R', on the same scope, the intersection of R and R', denoted $R \cap R'$,is the relation containing all tuples that are in both R and R'.
- The union $R \cup R'$ is the relation containing all the tuples that are in either $RorR'$ or both.
- The difference $R - R'$ is the relation containing those tuples that are in R but not in R'.
- Refer to your exercises (number 3).

Definitions (1/2)

- Selection. A selection takes a relation R and yields a new relation: the subset of tuples of R with specified values on specified variables. Example of notation: $\sigma_{x_1=a_1,\cdots,x_k=a_k}$
- Projection. Projection takes a relation R and yields a new relation that consists of the tuples of R with certain components removed. Example of notation: $\pi_{\{x_1,x_2\}}$.

Definitions (2/2)

- Join. The join operator takes two relations R_S and R_T and yields a new relation that consists of the tuples R_S and R_T combined on all their variables in S and T. A tuple r is the join of R_S and R_T, denoted $R_S \bowtie R_T$, if it can be constructed as follows: (a) Take a tuple s from R_S, (b) Select a tuple t from R_T such that the components of s and t agree on the variables that R_S and R_T have in common, and (c) Form a new tuple r by combining the components of s and t, keeping only one copy of those components correspoding to variables in $S \cap T$. Refer to your exercises (number 4).

Definitions (1/1)

- SAT is a decision problem considered in complexity theory.
- An instance of the problem is a Boolean expression written using only AND, OR, NOT, variables and parentheses.
- Question: given the expression, is there some assignment of *TRUE* and *FALSE* values to the variables that will make the entire expression true?

$$\neg A \vee \neg B$$
$$B \vee C$$
$$\neg C \vee \neg D$$
$$D \vee E$$
$$A \vee E$$

Thanks!

- Questions? or...
- Fancy coffee!