Symbolic Artificial Intelligence (COMP3008)
Formulations (supplementary material for discussions during the lectures)
20 October 2025

Daniel Karapetyan
daniel.karapetyan@nottingham.ac.uk

Examples of Decision Problems

Pigeon Hole Problem

Given:

- n pigeons
- m holes

Need to place all the pigeons in holes

Constraints:

- At most one pigeon in a hole
- A pigeon can only be placed in one hole

Vertex Colouring Problem

We are given an undirected unweighted graph $G = (V, E)$, where V is the set of nodes and E is the set of edges. We are also given a number k of colours.

The problem is to find a proper vertex colouring, i.e. an assignment of colours to the nodes V such that no two adjacent vertices share the same colour.

List Colouring Problem

We are given an undirected unweighted graph $G = (V, E)$, where V is the set of nodes and E is the set of edges. Also, we are given a set $L(v)$ of colours allowed for each vertex $v \in V$.

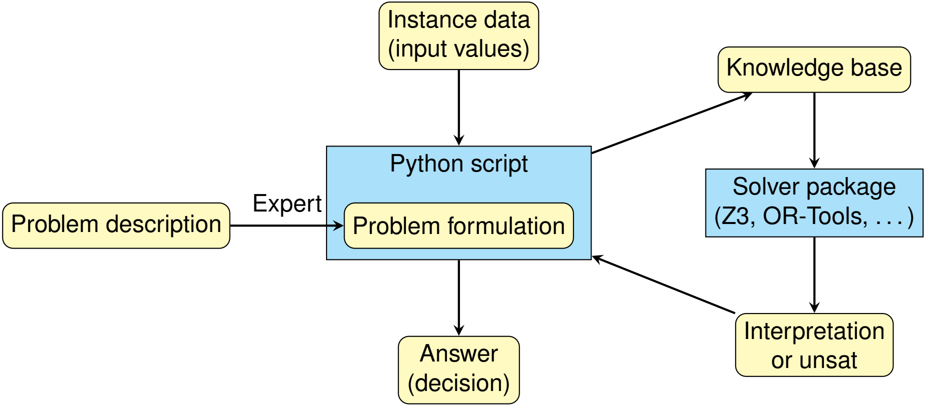
The problem is to find a proper vertex colouring such that, for each $v \in V$, the colour of v is in $L(v)$.

Exam timetabling problem

A university is timetabling exams. They have a number of rooms and time slots. They also know which exams each student takes. The goal is to timetable the exams so that no student has to sit two exams at the same time and no two exams happen in the same room at the same time.

Problem Formulations (from Lecture 3)

Real-world knowledge-based systems for decision support



Formulation

- Mathematically describes the problem
- Consists of constraints (knowledge); the solution has to satisfy all the constraints
- Parametrised: one needs instance data to produce a specific knowledge base
- There can be multiple correct formulations of a problem
- The performance of the solver may significantly depend on the chosen formulation

Formulation constraints (in FOL)

Single constraint:

$$\langle \text{FOL formula} \rangle$$

- Example:

$$\exists x . A(x) \rightarrow B(x)$$

Family of constraints – create a constraint for each described combination of parameters:

$$\langle \text{parametrised FOL formula} \rangle \qquad \langle \text{combinations of parameters} \rangle$$

- Example:

$$\exists x . A(x) \rightarrow B(y) \qquad \forall y \in Y$$

- You can use any unambiguous logic when describing the combinations of parameters

Formulation formatting

- Align the formulas
- Align the combinations of parameters
- Number the formulas to be able to reference them (even if you don't reference them yourself)
- A formulation can be split into multiple parts with the text
- Real-world FOL formulations will tend to also define the domain of discourse.

Example:

$$A(x) \qquad \forall x = 1, 3, \ldots, n-1 \tag{1}$$

$$\neg A(x) \qquad \forall x = 2, 4, \ldots, n \tag{2}$$

$$\forall x . B(x) \rightarrow A(x) \tag{3}$$

Designing Formulations

‘For all’ quantifier in formulations

In a mathematical formulation of a problem, we can write

$$A(x) \to B(x) \qquad \forall x \in X$$

Question: is this equivalent to

$$\forall x . A(x) \to B(x) \quad ?$$

‘For all’ quantifier in formulations

In a mathematical formulation of a problem, we can write

$$A(x) \to B(x) \qquad \forall x \in X$$

Question: is this equivalent to

$$\forall x . A(x) \to B(x) \quad ?$$

'For all' quantifier in formulations

In a mathematical formulation of a problem, we can write

$$A(x) \to B(x) \qquad \forall x \in X$$

Question: is this equivalent to

$$\forall x . A(x) \to B(x) \quad ?$$

How about this?

$$\forall x \in X . A(x) \to B(x)$$

‘For all’ quantifier in formulations

In a mathematical formulation of a problem, we can write

$$A(x) \to B(x) \qquad \forall x \in X$$

Question: is this equivalent to

$$\forall x . A(x) \to B(x) \quad ?$$

How about this?

$$\forall x \in X . A(x) \to B(x)$$

Correct equivalent in FOL:

$$\forall x . P(x) \to \big(A(x) \to B(x)\big), \tag{4}$$

$$P(x) \qquad \forall x \in X, \tag{5}$$

$$\neg P(x) \qquad \forall x \in \mathcal{D} \setminus X, \tag{6}$$

where $\mathcal{D}$ is the domain of discourse.

Quantifiers in formulations

Is this

$$\forall x.\exists y.A(x, y)$$

equivalent to

$$A(x, y) \qquad \forall x \in \mathcal{D},\ \exists y \in \mathcal{D},$$

where $\mathcal{D}$ is the domain of discourse?

Quantifiers in formulations

Is this

$$\forall x . \exists y . A(x, y)$$

equivalent to

$$A(x, y) \qquad \forall x \in \mathcal{D},\ \exists y \in \mathcal{D},$$

where $\mathcal{D}$ is the domain of discourse?

Correct equivalent:

$$\bigvee_{y \in \mathcal{D}} A(x, y) \qquad \forall x \in \mathcal{D},$$

where $\mathcal{D}$ is the domain of discourse

Input vs output of the solver

Graph colouring solver input:

- Number of nodes
- Set of edges
- Number of colours

You have to introduce notations for all the inputs

Graph colouring solver output:

- The assignment of colours to the nodes of the graph

Your solution representation is what describes the output:

FOL: predicates and functions such that their interpretation can be translated into this assignment

CSP: variables such that their interpretation can be translated into this assignment

How to use input data inside the formulation

Option 1

Use the inputs outside your FOL/CSP formula, e.g.

. . .

Let E be the set of edges.

. . .

$$c_v \neq c_u \qquad \forall (v, u) \in E\,.$$

You can even write it like this:

For every edge (v, u):

$$c_v \neq c_u\,.$$

(This way you do not need to introduce the notation for the set of all edges)

How to use input data inside the formulation

Option 2 (mainly relevant to FOL)

Create an FOL predicate/function and restrict its interpretation based on the input data, e.g.

> *...*
>
> *Let V be the set of nodes and E be the set of edges.*
>
> *...*

$$E'(v,u) \qquad \forall (v,u) \in E\,, \tag{7}$$

$$\neg E'(v,u) \qquad \forall v,u \in V \text{ such that } (v,u) \notin E\,. \tag{8}$$

$$\forall v . \forall u . E'(v,u) \rightarrow c(v) \neq c(u)\,. \tag{9}$$

Note that I cannot use E as a predicate inside FOL; FOL does not have a mechanism to define the interpretation of a predicate or function. Thus, I have to introduce a predicate E' and restrict its interpretation based on the input data E.