Session 8 SMV: N-Queen

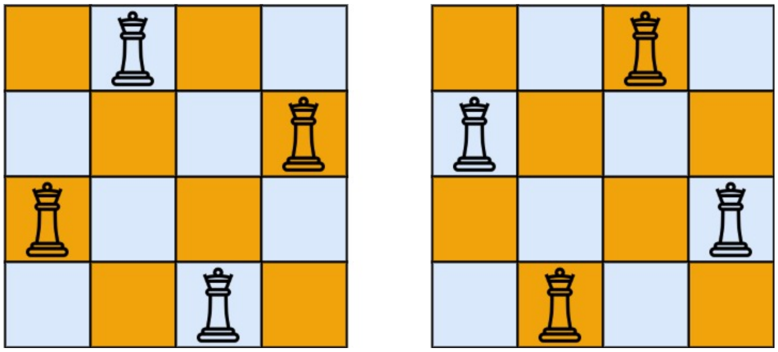
Damien Morard

16 December 2021

1 Solve the N-Queen problem using SFDD

The N-Queen problem consists in placing N queens on a chessboard of size $N \times N$ such that all queens cannot attack each other. For a more detail explanation, see Wikipedia. Thus, the purpose of this exercise is to create an algorithm to build the SFDD of all possibilities.

For instance, below is a solution for the problem with 4 queens.



1.1 Key question

1. What is a state in the N-Queen problem ?
2. How could you represent a state using a SFDD ?
3. What would be the keys ? Think to define a key order.

1.2 Construction of a first SFDD

In general, the efficient way to built all correct solutions is to start by enumerating all possibilities. However, the SFDD construction strongly depends on your problem. There is no unique solution, even though some are better.

1. Find a way to construct all possibilities. Draw your intuition ! Hint 1: You should use the number of queens as a constraint ! Hint 2: There is only one queen in each row.
2. Write an algorithm using pseudo-code.

1.3 Homomorphism to filter wrong solutions

The last step is to reduce solutions from the last exercise and keep only the good ones. Depending on the constraints that you have already introduced before, your list of homomorphisms can be different.

1. What are the constraints you must introduce to filter wrong solutions ?
2. Create a homorphism for each of these constraints. Use the same format as the one in the course (e.g.: union, intersection, insertion...) Hint: Think of what you need to compute the result !