Abstract
Many puzzles (single-played games) are available nowadays: e.g. Kakuro, Nanograms, Nurikabe, and Sudoku. The aim for each of them is to fill the grid with numbers or other objects by following a particular set of rules. We focus here on the Shirokuro puzzle, which in Japanese means "white-and-black". The purpose of this project is to develop a strategy for solving Shirokuro puzzles and to create the solver-program by means of the constraint programming.

Puzzle Analysis
The analysis of a few Shirokuro examples reveals interesting and useful restrictions on the puzzles.

I. The puzzle cannot contain the "diamond" group of circles. Such combination violates rules 2 and 3 of the game: the connection between the circles of one color will result in the isolation of the circles of the other color.

II. The outmost circles cannot form more than two groups of alternating colors. The reason is similar to the one of restriction I: the connection of circles of one color separates the circles of the second color.

III. It is worthwhile to include additional restrictions to a model. However, some of those restrictions are impossible to describe in the terms of the constraints provided by Choco. Other constraints would require multiple repetition for the different parts of the field. Consequently, the model will be overloaded. A few examples of groups that render the puzzle unsolvable are shown on the left. (Circles represent possible isolated groups.)

Strategy
A good start in solving the puzzle is the building the perimeter, wherever possible by filling in the space between contiguous circles of the same color with circles of that color. If contiguous circles have different color then the space between them remains empty. Building the perimeter significantly decreases the puzzle solving time. It also allows detection of unsolvable puzzles before investing too much time in solving them. Next step is adding "obvious" circles that would prevent formation of "diamond" groups and 2x2 same color groups. Afterwards, the model is created and submitted to the Choco solver. Every solution found by the Choco solver required a verification to identify the presence of isolated groups.

Current state of the project
The current version of the solver-program is able to either solve the puzzles of any reasonable size if the puzzle is solvable, or to determine that the puzzle has no solution. Simple GUI is provided. It allows for the drawing of the puzzle and the display of the solution. For testing purposes, GUI is limited to the 8x8 fixed size and displays only the first solution found.

Plans for the future
The main goal for the future would be to develop an advanced constraint that would allow for the dealing with the isolated groups.

The functionality of the GUI shall be extended with abilities to save desired puzzles on the disk, to load previously saved puzzles, and to browse the list of solutions.

Tools
The solver-program is based on the Java library (Choco) developed for constraint programming. This library can be used as a black box, i.e. all we need is to define the problem without knowing the solving strategy. The model is one of the key elements of any Choco program. It allows for a very simple formulation of a problem: the definition of a set of variables (which are the cells of the puzzle) and a second set of constraints of the problem (which are the rules of the puzzle). Detailed documentation on the Choco library could be found on its official website: http://www.emn.fr/x-info/choco-solver/.

Can you solve it?