

# Social Computing Article\*

Tabajara, Lucas

Institute of Informatics

Federal University of Rio Grande do Sul

Brazil

lmtabajara@inf.ufrgs.br and Prates, Marcelo

Institute of Informatics

Federal University of Rio Grande do Sul

Brazil

morprates@inf.ufrgs.br

## Abstract

Abstract

## 1 Introduction

Introduction

## 2 Background

Background

## 3 Contribution

Contribution

## 4 Methodology

We developed a system in which the environment of the sudoku problem-solving social network presented in [?] could be modelled through autonomous agent networks and analysed through a series of experiments. The optimization technique of *Memetic Networks* [?] was employed on the modelling, giving a basis for the dynamics of the agent network. A *Memetic Network Algorithm* is composed of an ordered set of  $N$  agents, each encoding a complete solution to the optimization problem, and a binary  $N \times N$  matrix encoding possible connections between agents. Additionally, a *Memetic Network Algorithm* is composed of a set of rules specifying how connections between agents are formed and erased and how interactions between these agents take place. These rules are grouped under the categories of *Connection Rules*, specifying how agents will connect and disconnect from each other; *Aggregation Rules*, specifying the dynamics of the information flow through connections; and *Appropriation Rules*, specifying how agents are supposed to add local changes to the information received through their connections.

Our solution adapts the *Memetic Network* technique - originally intended for use in optimization problems - to the context of constraint satisfaction problems. In this specific case (sudoku solving), we treat problems with one unique solution. To deal with this scenario, we propose a *Memetic Network* variation in which each agent encodes not a complete, but a partial solution to the problem.

Our solution employs a fixed topology for the agent network and models cooperation through a set of *aggregation rules* and reasoning through a set of *appropriation rules*. Our aggregation rules specify how sudoku partial solutions are copied from agent to agent, while our appropriation rules specify how agents increment these solutions with sudoku solving techniques such as naked singles, swordfish, etc.

### 4.1 Aggregation Rules

The experiments conducted by [?] point out a series of observations about the dynamics of cooperation in problem solving with human beings. For instance, the authors analysis has shown that users tend to copy not the most complete solutions, but instead the first solutions on the graphic interface. Additionally, we know from that same paper that users rarely copy solutions from more than 6 neighbours. In order to analyse and compare this behaviours, we developed a collection of aggregation rules, each modelling a copying strategy.

#### Pick Most Filled

This rule employs the intuitive strategy of copying from the neighbour with the most filled sudoku, namely the most complete solution

#### Pick Among First

We know that users tend to copy the first (from left to right) solutions on the graphic interface [?]. The authors have provided us with a mathematical model of this behaviour, stated as  $X(k) = (1 - p)^{k-1}p$ , where the parameter  $p$  is fixed as  $p = 0.5479$  and  $X(k)$  denotes the probability of an agent copying the  $k_{th}$  neighbour solution.

We inserted this behaviour into our model by firstly generating a random ordering of neighbours for each agent to compose a simulated graphic interface in which each agent visualizes some neighbours before or after others. Secondly, we translated the above mathematical model into an aggregation rule in which the solution copied by an agent is the  $k_{th}$  solution with probability  $X(k)$ .

### 4.2 Appropriation Rules

In sudoku strategy literature, as in chess, we find multiple techniques with iconic names. Some popular examples are *naked singles*, *naked twins*, *swordfish*. These strategies intend to, given a sudoku puzzle in a partial state of completion, generate movements to mark blank cells of the puzzle, as "mark

---

\*This title is temporary

*cell 1 of column 3 with value 5*". We reproduced a variety of these strategies, modelling each one of them as a function mapping a sudoku partial solution to a set of movements.

**Only Choice**

**Single Possibility**

**Two out of Three**

**Naked Twins**

## **5 Results**

Results

## **6 Conclusions**

Conclusions