Proceedings of the First AAAI Conference on Human Computation and Crowdsourcing

Leveraging Collaboration: A Methodology for the Design of Social Problem-Solving Systems

Lucas M. Tabajara, Marcelo O.R. Prates, Diego V. Noble, Luis C. Lamb
Institute of Informatics
Federal University of Rio Grande do Sul
Porto Alegre, RS, Brazil
lmtabajara@inf.ufrgs.br morprates@inf.ufrgs.br dvnoble@inf.ufrgs.br lamb@inf.ufrgs.br

Abstract

Social collaboration has been shown to facilitate problem-solving activity in diverse sets of environments. Nevertheless, if not well designed, social and human computation systems may achieve results only similar to those of a single human subject performing a task. This scenario reflects a need for better understanding of the performance issues of human problem-solving social networks. Firstly, we propose a model for simulating social problem-solving. We then carry out several simulations with artificial agents supported by results of experiments carried out with human subjects, in order to analyse which parameters influence the performance of collaborative problem-solving social networks. We analyse the strategies humans follow when solving a problem, comparing them with alternative ones, and identify the consequences of the employed strategies in the collective performance of the social network. Our results also indicate that copying and guessing are beneficial to the performance of the social networks. We then propose mechanisms that can improve collaborative problem-solving. Finally, we show that our results lead to a methodology for the design of efficient problem-solving systems that can be applied to several kinds of collaborative social systems.

1 Introduction

Problem-solving has been the subject of intense investigation since the dawn of Artificial Intelligence (Newell *et al.* 1959; Simon 1990). Recently, there has been an increasing interest in the study of human beings as (collaborative) problem-solving agents (Law and von Ahn 2011; Nagar and Malone 2011; Woolley *et al.* 2010). Such investigations have shed new light on the way humans interact to solve problems as well as on the dynamics of working groups (March 1991; Clearwater *et al.* 1991; Nagar and Malone 2011; Bernstein *et al.* 2012). However, although these studies can provide us with observations and hypotheses, there is still much to explain about the observed collective, human behaviour (Simon 1990). Thus, in order to better employ, apply or model human problem-solving cognitive abilities we need a method for studying human behaviour and its consequences (Mason and Watts 2011).

Humans are known to easily perform tasks which are still generally difficult for computers, such as natural language communication or image recognition. Modelling human abilities may thus lead to novel approaches or insights to computational problem-solving techniques that might even be more effective than current computational approaches. Recently, human problem-solving abilities have been applied to several problems, e.g. the protein structure prediction problem (PSP) in bioinformatics. The success of the *Foldit* game (Khatib *et al.* 2011) led to significant results in PSP, which is usually solved by optimization algorithms requiring intense computing power. Such results have been attributed to human visual problem-solving and decision-making abilities, but also to social collaboration (Khatib *et al.* 2011). However, we still do not know the limits of human abilities in problem-solving and how they compare to more traditional computational techniques. In order to take full advantage of human problem-solving abilities, we must learn their limitations. Humans are less than effective in mathematical computation, they are subject to e.g. physical and psychological conditions that affect their performance, and do not always act rationally (Simon 1990; Nagar and Malone 2011; Bernstein *et al.* 2012). In this paper, we shall use multi-agent based simulation to complement the study of social computation, in order to explain the strategies used by humans when solving problems and understanding their consequences in a collaborative environment. Such simulations draw inspiration from empirical results of Farenzena et. al.(Farenzena *et al.* 2011) which in turn conducted experiments with human subjects in a social computing environment. By using our model, it becomes possible to draw new conclusions from past observations of human behaviour. The model can be used, e.g. to preview the results that changes in the infrastructure of a social computation system will have on its overall performance before actually performing the experiments with humans. Further, one may use your methodology in the design of social computing systems.

In summary, we introduce a novel method for artificial social problem-solving that can be used to simulate the behaviour observed in humans in collective problem-solving systems. Section 2 presents related work. Our problem-solving model is described in Section 3. The methodology is then applied to simulate a social problem-solving system in Section 4. We shall simulate human behaviour observed in previous experiments and compare human problem-solving strategies with an alternative, artificial strategy. We show that human strategies may be detrimental to problem-solving process. A more detailed analysis is provided in Sections 5 and 6. In addition, we analyse the impact that copying and guessing have on the process of solving the problem. Our results contribute towards the design of effective social problem-solving systems.

2 Background

Human and social computation are relatively new research fields founded on diversified, interdisciplinary areas including the social sciences, artificial intelligence, game theory and network science (Hogg and Huberman 2008; Easley and Kleinberg 2010). Recent studies on the potential of human social networks as problem-solving tools have provided insights into, among other things, the impact of network structure in the collaboration process and the factors that lead agent's neighbours proposed solutions to be copied by their peers (Nagar and Malone 2011; Kearns 2012; Rendell *et al.* 2010).

The origins and foundations of human computation can be traced back to the work of (Dawkins 1986), in which the evolution of two-dimensional sets of line segments was guided by the aesthetic perception of human subjects. Nowadays, the use of human evaluation as a component of the fitness function in genetic algorithms is known as *interactive evolutionary computation*. However, it is interesting to note that already in the 1930s people were used as "computors", as noted in e.g. (Turing 1936). Recently, (von Ahn and Dabbish 2008), identified the possibility of using entertainment as an incentive to participation of human subjects, applying it in games in which the participants are actually performing a computation. That is an idea that also appears in the Foldit game (Khatib *et al.* 2011).

The series of experiments summarized in (Kearns 2012) are among the first to try and take advantage of collective problem-solving abilities to solve classical computer science problems. Such experiments are mostly based on the concept of coordination: subjects have individual incentives that are expected to drive them to cooperate with one another and lead them toward the collective goal (Nowak 2006).

Other initiatives have appeared, such as the ones by (Farenzena *et al.* 2011) and (Mason and Watts 2011), which take a different approach by having subjects trying to solve the collective problem individually, with the possibility of exchanging solutions between neighbours. Those have resulted in interesting conclusions on human behaviour when the possibility of copying peers is available. Our experiments build upon this line of research.

3 A Novel Method for Social Problem-Solving

Several multi-agent methodologies employ information flow through agents (Montanari and Saberi 2010; Villatoro *et al.* 2011). For instance, the model proposed by (Araujo and Lamb 2008) draws inspiration from the phenomenon of *Cultural Evolution* discussed by Dawkins (1976) and has a network of agents sharing, copying and incrementing units of information in a similar way that nature, according to Dawkins, deals with *Memes* (Dawkins 1976). Nevertheless, some aspects of social behaviour of great significance to social computing cannot be properly analysed through these methodologies. One example is the conformist behaviour studied in (Efferson *et al.* 2008). The proper study of these aspects demands a novel method for artificial collaborative problem-solving.

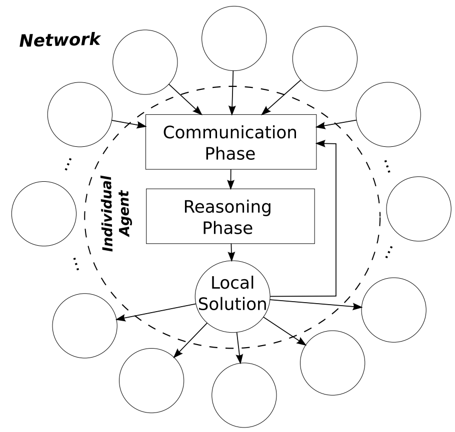

Figure 1: Simplified method diagram. Note that the reasoning phase output is stored as the agent's local solution and that the set of agents that contributed in the communication phase is not necessarily disjoint from the set of agents to which the agent's local solution is to be broadcast.

We propose a method for solving computational problems by means of a network of agents endowed with social behaviour. Our method is represented in the simplified diagram of Figure 1. Note that reasoning phase output is stored as the node's local solution and that the set of agents that contributed in the communication phase is not necessarily disjoint from the set of agents to which the individual's local solution is to be broadcast.

Our model employs a meta algorithm, named MASP (Meta-Algorithm for Social Problem-Solving), which considers an ordered set of N agents, each encoding a partial solution to the problem, and a binary $N \times N$ matrix representing possible connections between them. In addition, the MASP algorithm represents two model stages, namely the *communication phase* and the *reasoning phase*.

In the *communication phase*, solutions are exchanged between agents through the network connections. Agents are thus presented with a multiplicity of messages. There is a particular probability associated with the behaviour of agents choosing to copy one of these solutions in contrast to keeping their current ones. We call this probability the *copy rate*. When an agent chooses to copy, it is then supposed to select for copying a single one of its received messages. This is done by means of a particular strategy. In the *reasoning phase*, agents are supposed to add local changes to the solutions copied in the previous stage.

Algorithm 1 MASP: Meta-Algorithm for Social Problem-Solving, encompassing the communication and reasoning phases

```
Initialize N agents, each encoding a partial solution to
the problem;
while termination condition not met do
    for i = 1 to N do
        for j = 1 to N do
            if j is connected to i then
                A_i = i_th agent;
                A_j = j_th agent;
                Add A_j's solution to the collection of
                messages received by A_i;
    for i = 1 to N do
        //Communication Phase ;
        A_i = i_th agent;
        selectedMessage = select(A_i.messages);
        if random(0.0, 1.0) < copyRate then
            A_i.solution = selectedMessage;
    for i = 1 to N do
        //Reasoning Phase ;
        A_i = i_th agent;
        Add local changes to A_i's solution
```

4 Modelling Collaboration

In order to validate our methodology we have modelled a real-world collaborative Sudoku solving environment and test it over a set of problem-solving instances.

4.1 Communication Phase

The experiments conducted by (Farenzena *et al.* 2011; Rendell *et al.* 2010) 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 human subjects are more likely to engage in the behaviour of copying the most readily available solutions on the graphic interface than in that of evaluating the available solutions and choosing the best one according to some criteria. That behaviour is referred to as an evidence of conformism in human subjects. On the other hand, (Mason and Watts 2011) seem to suggest that their subjects did evaluate the available solutions, although an in-depth analysis of that fact was not reported. That might mean that multiple factors might influence the behaviour of human agents. We analyse possible reasons in Section 6.

Based on the experiments of Farenzena et. al., we have modelled two alternative message-selecting strategies in the communication phase; the first one attempts to evaluate the solutions and chooses the best one, while the other mimics human behaviour by selecting the first available solutions with greater probability. Although we employ such strategies in the context of the Sudoku puzzle, these strategies can be applied to different problems. (Farenzena *et al.* 2011) report that copying arbitrary solutions is not strictly dependent on the problem.

***Selective* strategy**: This strategy selects the most complete available solution for copying. In the specific case of Sudoku solving, this strategy selects the Sudoku partial solution with the largest number of filled in cells. This strategy evaluates the solutions available to the agent based on their content. It is important to notice, however, that it does not guarantee that the chosen solution is actually better; in the case of Sudoku, a cell might be filled in with a wrong value.

***Positional* strategy**: In some settings of collaborative problem-solving, human subjects are likely to copy the first (from left to right) solutions available on an HCI interface (Farenzena *et al.* 2011). This behavior is modelled by $\langle X(k)\rangle = (1-p)^{k-1}p$, where p is fixed as $p = 0.5479$ and $\langle X(k)\rangle$ denotes the probability of an agent copying the k_{th} neighbour solution. In order to simulate a graphic interface we generate a random ordering of neighbours for each agent. Thus each agent visualizes its neighbours in a specific order. Secondly, we translated the above mathematical model into a solution-selecting strategy in which the an agent selects the k_{th} solution with probability $\langle X(k)\rangle$.

4.2 Reasoning Phase

The techniques employed by the agents to solve a problem individually are chosen according to the problem in question. In the case of Sudoku, problem-solving techniques abound in the literature (Weber 2005). These techniques are based on the reasoning usually employed by humans when solving the puzzles. (Davis 2011) discusses a collection of such techniques: e.g. the *Naked Singles*, the *Hidden Singles* rule and the *Naked Twins* rule. These techniques intend to, given a partial Sudoku solution, generate a set of *movements* which can be used to mark cells of the Sudoku puzzle. We implemented five rules, modelling them as functions that map Sudoku partial solutions to a set of movements. These functions can be used in the reasoning phase to add local changes to the Sudoku solution received in the communication phase.

The rules implemented were the *Unique Missing Candidate*, the *Naked Singles*, the *Hidden Singles*, the *Two out of Three* and the *Naked Twins* rule, discussed in (Davis 2011). In our modelling, each agent knows a particular quantity of rules, this quantity determining the agent's *level*. For instance, an agent of level 3 knows three out of the five rules. Agents of level 0 know no rules and therefore can only guess and copy. That way, we model a wide range of skills that might be found in human agents. Our tests were all conducted with a heterogeneous population of agents of different levels.

Guessing and Backtracking: We have consistent evidence that trial-and-error is a part of the Sudoku solving experience. The need for trial-and-error in Sudoku puzzles is not a falsifiable conclusion, but a mathematical fact (Davis 2011). Some puzzles are only solvable by the means of a backtracking procedure.

	Number of agents						
	By level						Total
Topology	0	1	2	3	4	5	
Scale-Free ($\gamma = 1.58$)	6	5	4	4	4	4	27
Ring ($k = 2$)	1	1	1	1	1	1	6
Ring ($k = 3$)	3	3	3	3	3	3	18
Fully-connected	4	4	4	4	4	4	24

Table 1: Topology configurations used in the experiments.

In our modelling, we associate each agent with a numerical parameter determining the probability this agent has to guess when incapable of applying a typical Sudoku solving strategy. We call this parameter the *guess rate*.

Automatic Sudoku solvers employing a backtracking algorithm are easily programmed and very time-efficient. On the other hand, the space complexity of these algorithms is a barrier to most human solvers, who need to write down tons of observations in order to employ a backtracking strategy. With this in mind, we propose in our modelling a different kind of backtracking, intended to be more similar to the way human beings employ error correction in Sudoku solving in a collaborative environment such as the one analysed in (Farenzena *et al.* 2011), which we refer to as *social backtracking*. In it, when faced with one or more conflicts in its own solution, an agent copies a solution from one of its neighbours. In our modelling, this is done by raising the copy rate of this particular agent to 1.0.

5 Experiments and Results

We modelled our agent networks using four different topologies: a *fully-connected* topology, a *scale-free* topology (Barabási *et al.* 2001) and two *ring* topologies, one with $k = 2$ and another with $k = 3$, where the degree of the network is $2 * k$ (Newman 2010). Table 1 shows details of each topology used. We implemented the Sudoku environment using the agent distributions shown in Table 1. We performed experiments with varying solution selection strategies, network topologies and values for the copy and guess rates. We have run experiments for seven different problem instances, and obtained similar results in each case.

Although different Sudoku instances have varying levels of difficulty, we have chosen to leave this parameter out of our experiments. Two reasons justify our decision: (1) there is no consensus in the literature about an objective quantitative difficulty measure for Sudoku, and (2) at this initial stage, we are more concerned with variables which are independent from Sudoku problem-solving itself. This choice can indeed allow us to generalize our results to other problems, including brain teasers.

5.1 Comparing Copying Strategies

We ran 10 experiments for each combination of topology and instance. For all runs, we fixed the values of copy and guess rate at 0.5. Each experiment ran for 100 rounds composed of a communication phase and a reasoning phase. We first run the *Selective* strategy and then the *Positional* strategy, with the goal of comparing their performance. Our hypothesis was that the *Selective* strategy would perform better, a result which would confirm our belief that problem-solving social networks respond positively to the employment of solution evaluation.

We have observed in our experiments that the progress of the solution, represented by the number of agents that solve

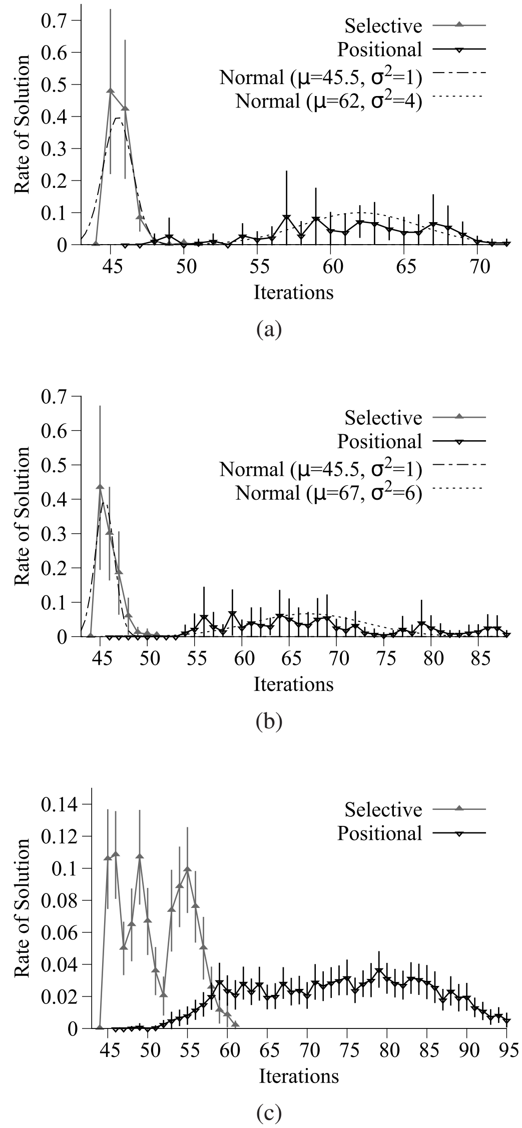

Figure 2: Graphs representing average results for (a) 3-ring; (b) scale-free; and (c) for all topologies and instances. The graphs show the proportion of agents obtaining the correct solution on each round for both the *Selective* and *Positional* strategies; error bars represent a 95% confidence interval. The lines for both strategies can be fitted to normal distributions with different values for mean and variance.

the problem in a given round, follows a normal distribution. During the first rounds no agents have found the solution. The round in which the first few agents solve the problem varies according to the particular instance. From that point onwards the final solution starts spreading throughout the network. As more agents obtain the complete solution, it becomes more likely that further agents will copy that solution from their neighbour, which complies with a conformist behaviour, as described by (Efferson *et al.* 2008). The number of agents with complete solutions increases in the following rounds, until it reaches a point where the few remaining agents take longer to obtain the complete solution.

The solution progress for the *Selective* and *Positional* strategies can be described as normal distributions of different mean and variance. In the experiments, the mean indicates how soon the correct solution was found, and the variance how long the final solution took to spread throughout the network. The exact values for these parameters vary according to factors such as topology and problem instance, but they follow a particular pattern: in all cases, the *Selective* strategy has lower mean - indicating the solution is found sooner - and lower variance - meaning it spreads faster.

We have chosen the scale-free and 3-ring topologies as representative examples, and plotted the average of the results for the same instance of the problem for each topology in Fig. 2a and 2b, respectively. Each graph shows the average of the percentage of agents that find the correct solution in each round for the two strategies, with normal curves plotted over the values evidencing the behaviour described above. The curves were tested for normality using *D'Agostino's K^2 Test* (Pearson *et al.* 1977), obtaining for the *Selective* strategy, for the interval $[40, 50]$, $(K^2 = 7.12, \chi^2 = 0.03)$ in the 3-ring topology and $(K^2 = 5.58, \chi^2 = 0.06)$ in the scale-free topology, and for the *Positional* strategy, for the interval $[50, 75]$, $(K^2 = 2.62, \chi^2 = 0.27)$ in the 3-ring topology and $(K^2 = 1.67, \chi^2 = 0.43)$ in the scale-free topology.

Fig. 2c depicts the average for all experiments, encompassing every topology and instance. In this graph, the presence of the normal behaviour in the distribution is less clear, because it combines several different instances and topologies. Nevertheless, its effects can still be seen. The *Selective* strategy displays different residual peaks from particular instances, while the line for the *Positional* strategy starts flattening with the accumulation of several distributions from the individual experiments. More importantly, Fig. 2c shows that the results of Fig. 2a and 2b can be generalized: in average, for every topology and instance, the network converged to the correct solution between rounds 44 and 60 with the *Selective* strategy, and between rounds 54 and 94 with the *Positional* strategy, showing that for the *Selective* strategy the solution is found earlier and spreads faster.

5.2 Varying Copy and Guess Rates

In the tournament conducted by (Rendell *et al.* 2010), the most successful strategies relied heavily in social learning. Based on that idea, we hypothesised that a higher copy rate would improve performance, while a higher guess rate would decrease it. We based such hypothesis on the notion that, when an agent makes a guess, it has a high chance of filling a cell with a wrong value. Once that happens, that grid will assuredly be unable to lead to the right solution until the agent copies a grid without errors from one of its neighbours. Therefore, we hypothesised that if an agent cannot make a logical move using its level of Sudoku-solving skills, it is better for the agent to wait for its neighbours to improve the solution and copy from them, instead of risking making an incorrect move.

In order to test our hypotheses, we repeated our experiments for the same four topologies, this time varying the copy and guess rates between 0.0 and 1.0 each. We plotted two surfaces for each strategy and topology. The results were similar for each topology, so we chose the scale-free topology as a representative example to present in Fig. 3a and Fig. 3b.

Fig. 3a and Fig. 3b show the average number of rounds needed for the network to converge to the correct solution for the *Selective* and *Positional* strategy, respectively. They both show a maximum in (copy rate: 0, guess rate: 0) and a minimum in (copy rate: 100, guess rate: 100), with greater copy and guess rates reducing the number of rounds needed for the agents to solve the problem. The results also differ depending on the copying strategy employed in the communication phase. Fig. 3a shows that, using the *Selective* strategy, as copying increases the number of rounds needed for all agents to have the correct solution decreases sharply until it stabilizes in a roughly optimal level. Similarly, in Fig. 3c the percentage of experiments in which every agent obtains the correct solution rises fast as copying increases. Meanwhile, the benefits of guessing are much less pronounced. On the other hand, when the agents do not evaluate their neighbours' solutions, as seen in Fig. 3b and Fig. 3d, their contributions do not differ as much.

6 Discussion

We now discuss results from the previous sections and analyse their consequences with respect to the design of collaborative problem-solving systems.

6.1 A New Model of Human Problem-Solving Networks

We have proposed a model of human behaviour in a problem-solving social network. In the model one can simulate previous experiments performed by humans by modelling the observed behaviour as the reasoning and communication phases. That way, one can examine the consequences of that behaviour without having to enlist human subjects, which is usually expensive and time-consuming. We have applied the model in a human computing experiment and reached particular conclusions about the strategies used by humans. Since the same behaviour was observed for different problems, we believe these results can be generalized to a certain class of problems and systems. We provide further analyses below.

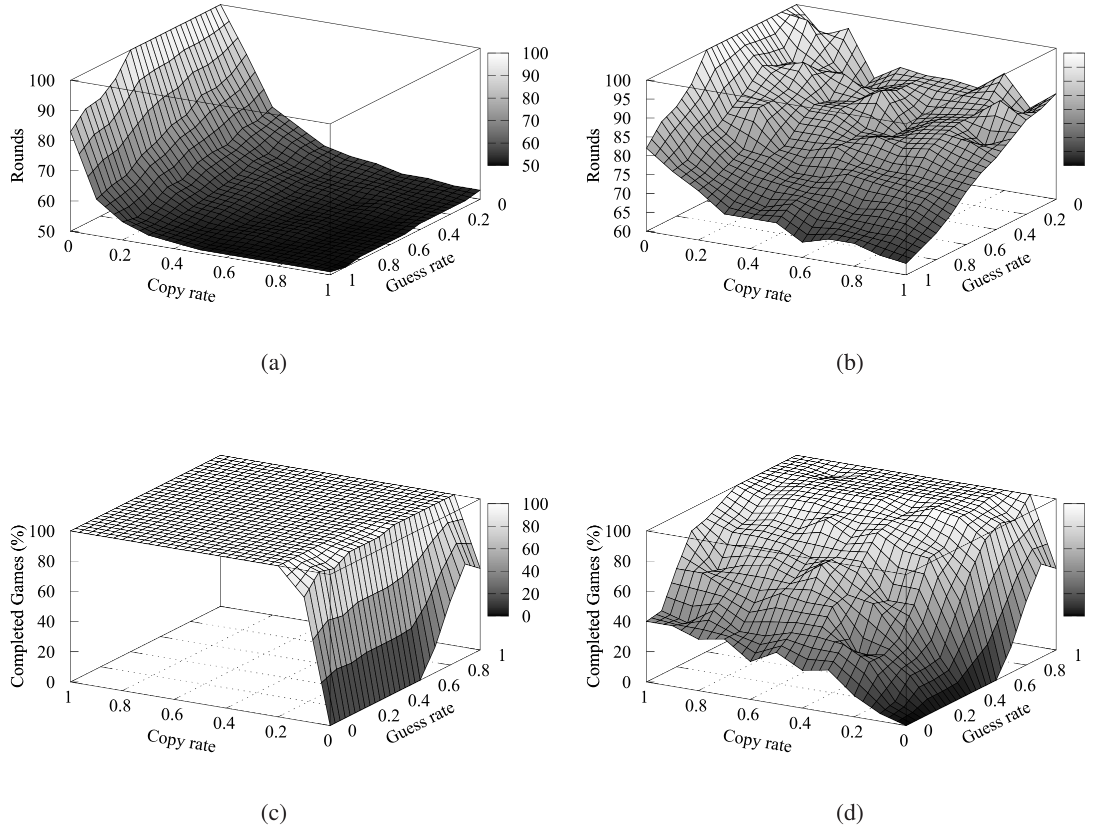

Figure 3: Figures (a) and (b) show the average number of rounds needed for all agents to obtain the correct solution, for a given value of copy rate and guess rate, when using (a) the *Selective* strategy; and (b) the *Positional* strategy, for the scale-free topology. Figures (c) and (d) show the percentage of experiments in which all agents obtained the correct solution, for a given value of copy and guess rates, when using (c) the *Selective* strategy; and (d) the *Positional* strategy, for the scale-free topology.

6.2 Evaluating Neighbours' Solutions is Advantageous

Our experiments confirm the hypothesis that *Selective* copying indeed brings better results. In all topologies, choosing the solution with the largest number of filled cells not only allows agents to solve an instance earlier, but also causes the network to converge faster to the correct solution. That is, once one individual has found the correct solution, that solution spreads faster through the network if the other agents are evaluating the solutions that reach them instead of copying an arbitrary one.

Although choosing a solution based on its position instead of content is provably inefficient, this behaviour has been observed in human networks (Farenzena *et al.* 2011). Perhaps this can be explained by humans finding it difficult to evaluate Sudoku partial solutions. We have used the number of filled cells as a solution evaluation function, but that is not guaranteed to accurately measure which solution is best, since individuals might fill in cells with a wrong value.

Our experiments, however, have shown that although completeness is not necessarily an accurate measurement of quality, it was good enough to lead the network to the right solution. These results might extend to other problems that do not have an accurate method of solution evaluation. Therefore, employing an imperfect solution evaluation method can be sufficient in problem-solving social networks.

6.3 Copying and Guessing Improve Network Performance

Regarding our hypothesis that a higher copy rate would be beneficial, our experiments indicate that in the topologies tested copying indeed has a decisive role in improving the network's performance. However, our results also indicate that guessing is not detrimental to the problem-solving, which is somewhat surprising. On the contrary, in several cases guessing actually contributes to improving the solution. We understand this is due to the fact that some low-level agents are incapable of solving some instances of the problem, ignoring the problem strategies. These agents have no hope of solving the problem individually without some guessing mechanism, and a high copy rate allows them to correct wrong guesses through copying better agents' solutions. What is particularly unexpected, however, is that the system did not seem to be detained by wrong guesses, even at high guess rates. This means that incorrect guesses were successfully filtered out, allowing the correct solutions to prevail. The agents seem to have been able to identify soon when a guess leads to an incorrect solution, eliminating it before it has a chance to spread.

Comparing the results of the different copying strategies, one can also observe that if the agents are evaluating their neighbour's solutions the copy rate has more influence over how fast the correct solution spreads through the network. When copying increases, the network performance also increases much faster for the *Selective* strategy than for the *Positional* strategy. At the same time, guessing has much less influence over the *Selective* strategy performance. These results confirm that the benefit of copying increases when the neighbours' solutions are evaluated. Meanwhile, when the agents do not evaluate the solutions they get less out of the act of copying, having to rely more on guessing to advance on their own.

6.4 On the Design of Social Problem-Solving Systems

Considering our results, social computing systems may take advantage of the interface to encourage human agents to select better-evaluated solutions. One possibility is to display the evaluation of each solution to encourage players to copy the ones with better score. Another possibility is using the very human behaviour of copying the first solutions to the system's advantage.

By ordering the neighbours' solutions in the interface according to their quality, with better-evaluated ones being displayed first, these will be copied more frequently. This strategy may prove to be highly beneficial to the overall performance of the system, given the fact that the computational cost associated with evaluating the quality of a solution may be negligible compared to the overall payoff of pushing the players toward the optimal path. It is important to highlight, however, that "first solutions" is a dangerous expression to use when we consider the whole spectrum of players' cultures and nationalities. The nature of the written language of a player may shape his/her perception of the first solutions: for example, certain populations, notably eastern players may perceive the rightmost solutions on the graphic interface as the first, given the right-to-left, top-to-bottom nature of some eastern languages.

There is also the possibility suggested of limiting the network's degree: fewer connections allow for all neighbours to be displayed in the interface at once, without the need for the user to search for solutions. Moreover, humans might feel more encouraged to evaluate the solutions if they have fewer options to choose from. That result is supported by (Mason and Watts 2011), who limited the number of agents' neighbours to three. In their experiments, the subjects seem to have indeed evaluated the neighbours' solutions, which might indicate this option to be promising.

Our results also show that a network benefits from higher copy rates. One way to achieve that is to take advantage of a phenomenon presented in (Mason and Watts 2011): the authors concluded that subjects copied more frequently when more than one neighbour shared the same solution. At the same time, higher local clustering in the network increased the probability that two neighbours of the same individual were also connected, which increased the likelihood that they shared the same solution. Therefore, by using topologies with higher clustering one can encourage human subjects to copy.

6.5 On a General Model for Social Problem-Solving

With respect to generalizing our model to other problems (such as logical problems in general, including brain teasers), as long as the social situation is similar to the one modeled in the paper, we believe that our results can be generalized. In particular, problems that can be framed as constraint satisfaction problems (including brain teasers) can be dealt with by our approach. Exploring whether our results generalize to problems other than constrain satisfaction problems remains an open question. We expect that in other real-world problems or scenarios, one may observe social effects such peer pressure, coalition formation, leadership, functional diversity, learning, exogenous centralized control among other effects. However, these effects shall demand specific investigations. We are interested in this paper in analyzing the results of a simple decentralized problem-solving system.

7 Conclusions and Further Work

Collaborative, social and human computing are growing areas of interest as researchers better understand human cognition and problem-solving abilities and strategies. However, although a lot of data about human social cognition can be collected from experiments, it is still difficult to explain human behaviour and its consequences (Lazer *et al.* 2009). Knowing the limitations of human strategies is needed in order to guide the design of social problem-solving systems. We have proposed a novel method for artificial social problem-solving that can be used to analyse the behaviour observed in humans in collaborative, social networks.

We have experimented with human strategies and obtained interesting results. Although humans seemed to choose a solution to copy based on its position on the interface, we have shown that evaluating the neighbours' proposed solutions leads to the correct solution being found sooner and spreading faster throughout the network even if the evaluation function employed is naive. Another important conclusion is that the performance of the network in problem-solving increases when agents copy more. Surprisingly, increasing the chance that agents will guess is also beneficial, indicating that the network is successful in filtering incorrect solutions produced by guessing.

With these results in mind, we have proposed some suggestions to improve the performance of a human social problem-solving network, using properties of the graphic interface and topology to encourage human subjects to engage in a more effective behaviour. These include displaying better-evaluated neighbours' solutions first - to encourage subjects to copy from them - and using topologies with higher local clustering - to encourage a higher frequency of copy. Future developers of collaborative and social systems can use these suggestions in their design in order to take maximum advantage of the problem-solving network. In the future, it would be interesting to validate these suggestions through further experiments with human beings. We also envisage the employment of the proposed method in simulating other experiments with human subjects, using the obtained results as guidelines to the design of new systems of social computation.

Acknowledgments This work is partly supported by the Brazilian Research Agencies CAPES and CNPq.

References

Ricardo M. Araujo and Luis Lamb. Memetic networks: analyzing the effects of network properties in multi-agent performance. In *AAAI'08*, pages 3–8, 2008.

Albert-László Barabási, Erzsebet Ravasz, and Tamas Vicsek. Deterministic scale-free networks. *Physica A*, 299(3-4):559–564, 2001.

Abraham Bernstein, Mark Klein, and Thomas W. Malone. Programming the global brain. *Commun. ACM*, 55(5):41–43, 2012.

S. Clearwater, B. Huberman, and T. Hogg. Cooperative solution of constraint satisfaction problems. *Science*, 254(5035):1181–1183, 1991.

Tom Davis. The mathematics of sudoku. In *Expeditions in Mathematics*. Mathematical Association of America, April 2011.

Richard Dawkins. *The Selfish Gene*, volume 32. Oxford University Press, 1976.

R. Dawkins. *The Blind Watchmaker*. Penguin Books, 1986.

David A. Easley and Jon M. Kleinberg. *Networks, Crowds, and Markets - Reasoning About a Highly Connected World*. Cambridge University Press, 2010.

C. Efferson, R. Lalive, P.J. Richerson, R. McElreath, and M. Lubell. Conformists and mavericks: the empirics of frequency-dependent cultural transmission. *Evolution and Human Behaviour*, 29:56–64, January 2008.

D.S. Farenzena, R.M. Araujo, and L.C. Lamb. Collaboration emergence in social networks with informational natural selection. In *IEEE Third Intl. Conf. on Social Computing*, pages 555–559, 2011.

Tad Hogg and Bernardo A. Huberman. Solving the organizational free riding problem with social networks. In *AAAI Spring Symposium: Social Information Processing*, pages 24–29, 2008.

Michael Kearns. Experiments in social computation. *Communications of the ACM*, 55(10):56–67, June 2012.

Firas Khatib, Seth Cooper, Michael D. Tyka, Kefan Xu, Ilya Makedon, Zoran Popovic, David Baker, and Foldit Players. Algorithm discovery by protein folding game players. In *Proceedings of the National Academy of Sciences*, volume 108, pages 1–5, June 2011.

Edith Law and Luis von Ahn. *Human Computation*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2011.

David Lazer, Alex Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, Tony Jebara, Gary King, Michael Macy, Deb Roy, and Marshall Van Alstyne. Computational social science. *Science*, 323(5915):721–723, 2009.

James G. March. Exploration and Exploitation in Organizational Learning. *Organization Science*, 2(1):71–87, 1991.

Winter Mason and Duncan J. Watts. Collaborative learning in networks. In *Proceedings of the National Academy of Sciences*, volume 2011, pages 764–769, November 2011.

Andrea Montanari and Amin Saberi. The spread of innovations in social networks. *Proceedings of the National Academy of Sciences*, 107(47):20196–20201, 2010.

Yiftach Nagar and Thomas W. Malone. Making business predictions by combining human and machine intelligence in prediction markets. In *ICIS 2011 Proceedings*, 2011.

Allen Newell, J. C. Shaw, and Herbert A. Simon. Report on a general problem-solving program. In *IFIP Congress*, pages 256–264, 1959.

M. Newman. *Networks: An Introduction*. Cambridge University Press, 2010.

M. A. Nowak. Five rules for the evolution of cooperation. *Science*, 314(5805):1560–1563, December 2006.

E. S. Pearson, R. B. D'Agostino, and K. O. Bowman. Tests for departure from normality: Comparison of powers. *Biometrika*, 64(2):pp. 231–246, 1977.

L. Rendell, R. Boyd, D. Cownden, M. Enquist, K. Eriksson, M. W. Feldman, L. Fogarty, S. Ghirlanda, T. Lillicrap, and K. N. Laland. Why copy others? Insights from the social learning strategies tournament. *Science*, 328(5975):208–213, 2010.

H.A. Simon. Invariants of human behavior. *Annual review of psychology*, 41(1):1–20, 1990.

Alan M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42:230–265, 1936.

Daniel Villatoro, Jordi Sabater-Mir, and Sandip Sen. Social instruments for robust convention emergence. In *IJCAI*, pages 420–425, 2011.

Luis von Ahn and Laura Dabbish. Designing games with a purpose. *Communications of the ACM*, 51(8):58–67, August 2008.

T. Weber. A SAT-based Sudoku solver. In *12th Intl. Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, pages 11–15, 2005.

Anita Williams Woolley, Christopher F. Chabris, Alex Pentland, Nada Hashmi, and Thomas W. Malone. Evidence for a collective intelligence factor in the performance of human groups. *Science*, 330(6004):686–688, October 2010.