Self-Adaptation in a Network of Social Drivers: Using Random Boolean Networks

André M. Machado and Ana L. C. Bazzan
PPGC / Instituto de Informática, UFRGS
Caixa Postal 15064
91.501-970 Porto Alegre, RS, Brazil
{ammachado,bazzan}@inf.ufrgs.br

ABSTRACT

One of the major research directions in adaptive and self-organizing systems is dedicated to learning how to coordinate decisions and actions. Also, it is important to understand whether individual agents' decisions can lead to globally optimal or at least acceptable solutions. Our long term approach aims at studying the effect of several types of strategies for self-organization of agents in complex systems. The present paper addresses simulation of agents' decision-making regarding route choice when random boolean networks are used as a formalism for mapping information coming from other agents into the decision-making process of each agent. It is thus assumed that these agents are part of a social network (for example acquaintances or work colleagues). Hence, part of the information necessary to decide can be provided by these acquaintances (small-world), or by route guidance systems. With this approach we target a system that adapts dynamically to changes in the environment, which, in this case, involves other adaptive decision-makers, a challenging endeavor. We compare our results to similar ones reported in the literature. Results show that the use of a relatively low number of boolean functions and few information from acquaintances leads the system to an equilibrium.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*

General Terms

Algorithms

Keywords

Artificial Intelligence, Multiagent Systems, Self-organizing System, Traffic Simulation, Adaptation, Game-Theoretic Approaches

1. INTRODUCTION

In adaptive and complex systems it is almost impossible to oversee the issue of learning how to coordinate. In particular, we focus here on a route choice scenario. This scenario can be populated either with human drivers (as currently), or with autonomous vehicles. These so called pods or autonomous guided vehicles will in the near future do most of the driving for us. In both cases, we assume specific degrees of vehicle automation, which makes the problem not only more interesting but also more challenging.

In the former case (human drivers), it is reasonable to assume that currently the level of automation in our vehicles already allows us to operate them in a different way as, for instance, a few decades ago. Sensors and actuators are ubiquitous in modern vehicles. This increases the awareness of the state of the environment, besides enabling a kind of communication that we are not (yet) familiar with.

Besides the operational level (breaking, security issues, etc.), for the purpose of this paper, we are more concerned with issues related with accessing and generating *information for increasing awareness and improving decision-making*. In fact, vehicles equipped with RFID tags generate huge amounts of information. On the other hand, vehicles equipped with GPS-based navigators provide the driver ways to make better decisions.

In the case of autonomous vehicles, it even goes beyond this level of automation. Such vehicles are able to communicate using car to car (C2C or V2V) communication thus sharing valuable information in order to make decisions regarding routes and so on.

In the rest of the paper we refer to drivers (or agents) but note that basically the same applies to autonomous driving. In [4] we outline other details related to planning and en-route choice that are possible in both cases.

No matter if we consider human drivers or autonomous vehicles, decision-making in this scenario is highly coupled with others' decisions, i.e., it is a matter of collective intelligence. In [15] Tumer and Wolpert enumerate the challenges that underlie this problem. The main one for our purposes is that decision-making at individual level is coupled with other agents' decisions. This is a well known case of minority games. We have dealt with such scenarios in the past: [3, 12, 5, 10, 6]. In most of the cases we investigate specific techniques to let drivers evolve their decision-making strategies. The first paper formulated route choice as a minority game in which drivers possess different strategies according to their personalities. The second and third ones have investigated the role of information in route choice. The last two papers have dealt with reinforcement learning in order to let agents find good strategies by exploration of the space of possibilities.

In some sense the present paper summarizes these approaches.

From the first we use the idea of sets of strategies (here grounded on the formalism explained below); from the second and third we use the idea of information generation and information sharing. Finally the idea of reinforcement learning also appears in the present paper as agents need to explore and later exploit the space of actions.

Specifically, here we use the well-known formalism introduced by Kauffman [8], the random boolean networks (RBN's), in order to let an agent decide which action to take when the decision is binary. This formalism is based on boolean functions that map K inputs to one output. An example of such a boolean function for $K = 2$ would be the AND function. Applied to the scenario of route choice this would roughly be like this: "I drive the highway A1 only if X *and* Y have driven there".

In summary, in the present paper we depart from the assumption that the adaptation mechanism only emerges out of an individual learning process. Rather, we consider that the agents are interacting in a social network and hence the actions of other agents must be considered. Given the fact that not all actions can be taken into account (as this would require an enormous amount of exchanged messages and would not even guarantee coordination in the end), here we assume that information is exchanged only among a small group of acquaintances. This exchange aims at: i) feed a network of boolean functions, and ii) to replace functions that do now work at individual level for others that have proven more successful.

The rest of this paper is organized as follows. The next section discusses related works and briefly reviews some background ideas on decision-making regarding binary route choice, when formulated as minority games. Section 3 explains how the RBN formalism works. In Section 4 we present details about our approach. In Section 5 we present the scenarios for simulation of route choice under several conditions, and give the results of these simulations. The last section summarizes the conclusions and outlines the possible extensions.

2. RELATED WORK

In transportation engineering, the traditional method of route assignment is static. Routes for each driver are determined at the initial phase of the traffic simulation, by using econometric formalisms that find an equilibrium in a macroscopic way. This means that the process is not self-adaptive from the point of view of individual drivers. In the last decades, microscopic models of demand assignment have become more popular though. The motivations for using them are manifold: increasing complexity of trips; heterogeneity can be represented; individual adaption and learning methods can be used; integration of interactions among agents; integration of other levels of decision-making such as mode choice; etc.

A number of publications suggest the application of intelligent agent architectures to different travel-related choice processes such as route and mode choice. Agent based approaches seem to be particularly relevant when networks are dynamic or when dynamic information is available. In this case, agent-based approaches support immediate reaction to the environment (including other drivers).

Next we review some of the literature on demand assignment, focusing on route choice.

The activity-based travel demand module of MATSim simulator – the so-called MATSim-T module [2] – has been used for large-scale cases, such as a simulation of the complete Zürich area. In MATSim-T each agent possesses a set of fully elaborated daily schedules, including details of the route choice to connect two (or more) locations. Schedules of all agents are simulated, evaluated using an elaborated fitness measure and optimized by means of genetic algorithms and mutation operators for adapting the plans.

The research team around T. Arentze and H. Timmermans have developed activity-based models for travel demand generation grounded on existing theories in psychology and economics. Their efforts contributed to a new version of the ALBATROSS (A Learning BAsed TRansportation Oriented Simulation System) system for modeling activity-based travel demand [1].

Panwei and Dia [13] use a fuzzy neural architecture where socio-economic parameters are represented as fuzzy variables for simulating decision-making about whether or not to keep their initial route decision when new information is available. Simulation results were empirically tested and validated.

A large number of works about the effect of information on route choice uses abstract scenarios. These abstract scenarios are mostly inspired by congestion or minority games. The basic idea is that agents have to decide simultaneously between two routes; those that select the less crowded one receive a higher reward. Agents' repeated decision-making is coupled to some adaptation or learning strategy so that the next choice is adapted to the reward feedback. Based on this, an equilibrium may be reached – a pareto-efficient distribution, or the Wardrop's equilibrium [17]. In terms of game theory this means that no agent can improve its reward or reduce its costs by switching routes without worsening at least another agent.

Examples of such abstract two route scenarios can be found in Bazzan *et al.* [3] or in Chmura and Pitz [7]. In the latter, a reinforcement learning scheme is used that aims at reproducing the decision-making of human subjects in a corresponding experimental study. Before this, a similar scenario was tackled by Klügl and Bazzan [11]. However the latter included a second phase in decision-making during which the initial decision is processed in order to give a forecast to all or part of the agents. Thus, the agents not only learn to select a route in the first phase, but also how to evaluate the information received in the second phase. One important result of this study was that a certain share of agents that ignore traffic forecast turned out to be necessary for an efficient agent adaptation.

In these game-theoretic scenarios, the reward of agents when selecting a route is calculated based on the number of all agents that selected that alternative. This is a very abstract view and does not resemble the actual dynamics of traffic situations. Objectives such as finding the appropriate form of information that leads to equilibrium states cannot be accomplished with such abstract scenarios. In order to do this Wahle *et al.* [16] discuss a microscopic simulation of the traffic flow based on the Nagel-Schreckenberg rules for a two-route scenario. Information about the traffic state was communicated to the drivers in order to support their route choice. Information was generated by floating cars that transmitted their individual travel time after having finished their trips. In this scenario the authors analyzed the effect of different qualities of information determined for example by the numbers of floating cars. In some settings (e.g., giving delayed information to drivers), harmful oscillations occurred regarding the numbers of drivers on each route. In subsequent studies, the same authors presented results about using other forms of information calculated from a macroscopic point of view, such as overall density, mean velocity, or information regarding trends. These ideas were later elaborated by Klügl *et al.* [12] where agents could learn which information was the most useful. In [6] Bazzan and Klügl showed that providing information can be also useful in the context of the Braess paradox.

Although the study of Yamashita *et al.* [18] did not aim at reproducing a particular real-world system, they have used simulation for testing the effect of traffic information. Their concept of "route information sharing" assumes that drivers communicate details about their planned route to an information server that broad-

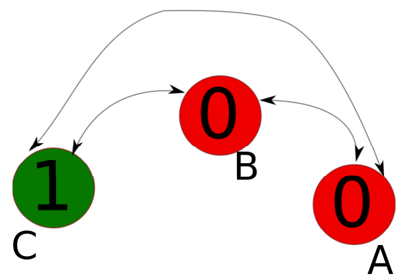


Figure 1: Example of a RBN with $N = 3$ connected agents ($K = 2$).

casts aggregated route information. They tested the system in various network topologies such as a grid or a radial ring structure. Their results showed that an increasing share of informed drivers yields decreasing travel times not only for drivers equipped with the information sharing device but also for the others. Moreover, travel times of drivers equipped with this device were lower than that of drivers without it. In 2009 this idea of sharing route information was further developed into a collaborative navigation system [19].

3. RANDOM BOOLEAN NETWORKS

Boolean networks have been used to explain self-organization and adaptation in complex systems. The study of the behavior of regulatory systems by means of networks of boolean functions was introduced by Kauffman in 1969 [8]. Examples of the use of this approach in biology, genomics, and other complex systems can be found in [9].

RBN's are made up of binary variables. In the setting investigated here, a network is composed of N agents that must decide between two actions. Each agent is represented by one of these binary variables. These in turn are, each, regulated by some other variables, which serve as inputs. The dynamical behavior of each agent, namely which action it will execute at the next time step, is governed by a logical rule based on a boolean function. These functions specify, for each possible combination of K input values, the status of the regulated variable. Thus, being K the number of input variables regulating a given agent, since each of these inputs can be either on or off (1 or 0), the number of combinations of states of the K inputs is 2^K. For each of these combinations, a specific boolean function must output either 1 or 0, thus the total number of boolean functions over K inputs is 2^{2^K}. When $K = 2$, some of these functions are well-known (AND, OR, XOR, NAND, etc.) but in the general case functions have no obvious semantics.

To illustrate the regulation process, Figure 1 depicts a simple example of a network of $N = 3$ agents where each was assigned a boolean function randomly, and $K = 2$. The boolean functions for these three agents are then depicted in Table 1 (adapted from [9]): agents A and B are regulated by function OR, while agent C is regulated by an AND. In this table, one can see all possibilities for C (3rd column) to make a decision, where 1 means for instance one route and 0 means another route. Similarly, A's output is determined by the inputs from both B and C, and B's output depends on inputs from A and C.

Given the three boolean functions from Table 1, Table 2 shows, for all 2^3 states at a given time T, the action taken by each agent at time $T + 1$, i.e., the successor state of each state. Further, from this table, it is possible to determine the state transition graph of the network, which appears in Figure 2. One sees that there are 3 state cycles (attractors).

(AND)			(OR)			(OR)		
A	B	C	B	C	A	A	C	B
0	0	0	0	0	0	0	0	0
0	1	0	0	1	1	0	1	1
1	0	0	1	0	1	1	0	1
1	1	0	1	1	1	1	1	1

Table 1: Boolean functions for agents C, A and B.

(T)			(T+1)		
A	B	C	A	B	C
0	0	0	0	0	0
0	0	1	1	1	0
0	1	0	1	0	0
0	1	1	1	1	0
1	0	0	0	1	0
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	1	1	1

Table 2: States' transition for Table 1.

If we assign each of the N agents randomly to one of the 2^{2^K} boolean functions, the dynamics of the decision-making is deterministic and the system ends up in one of the state cycles. It is then a matter of "luck" that only a certain fraction of agents end up using one route. For instance in the case depicted in Figure 2, in both state cycles 1 (`000`) and 3 (`111`), either none use one route (cycle 1) or all do (cycle 3).

Let us consider an example in which the boolean function of agent C changes from AND to NAND. The boolean functions are now depicted in Table 3 while Table 4 shows the successor state of each state and Figure 3 depicts the state transition graph.

Comparing Figure 2 and Figure 3, the dynamics of the regulation changes. Now only one state or attractor exists (`110`), namely one that has the property that agents A and B always go but agent C never does.

The extent of such a change – whether or not the system will be attracted to another attraction basin – obviously depends on the extent of the changes in the network and/or functions.

In [9] the author extensively discusses many of these factors, as well as the properties of RBN's, including the issue of stability and cycle length. In the present paper, because the logic of the functions and the structure of the network changes along time, properties such as periodic behavior change.

On the other hand, a central question raised by Kauffman, and

	↶ 000				state cycle 1
010	⇆	100			state cycle 2
		001			
		↓		↶	state
011	→	110	→	111	cycle 3
		↑			
		101			

Figure 2: States' transition graph for Table 1 (3 state cycles and attractors).

(NAND)			(OR)			(OR)		
A	B	C	B	C	A	A	C	B
0	0	1	0	0	0	0	0	0
0	1	1	0	1	1	0	1	1
1	0	1	1	0	1	1	0	1
1	1	0	1	1	1	1	1	1

Table 3: Mutated version of Table 1.

(T)			(T+1)		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1	1	1	1
0	1	0	1	0	1
0	1	1	1	1	1
1	0	0	0	1	1
1	0	1	1	1	1
1	1	0	1	1	0
1	1	1	1	1	0

Table 4: States' transition for Table 3.

$$\begin{array}{ccccccc|l} & & 000 & \rightarrow & 001 & & & \\ & & & & \downarrow & & \curvearrowleft & \text{state} \\ 010 & \rightarrow & 101 & \rightarrow & 111 & \rightarrow & 110 & \text{cycle 1} \\ & & & & \uparrow & & & \\ & & 100 & \rightarrow & 011 & & & \end{array}$$

Figure 3: States' transition graph for Table 3 (single state cycle and attractor).

which is relevant to our work, relates to the problem of adaptation in a system with many interacting parts. The key question is whether an adaptive process which is constrained to a change in the input connections between elements in a network and the logic governing them can hill climb to networks with desired attractors.

4. APPROACH

Basically the abstract scenario used in [10, 7] (and that we continue using here) can be seen as an special instance of the minority game: N agents repeatedly have to decide between two alternative actions. Let us assume that one alternative, namely M (main) is preferred (e.g., it provides more capacity). The other is a secondary one (thus S). At the end of the round, every agent gets a reward that is computed based on the number of agents who selected the same alternative. This mimics the actual travel time experienced by the agent itself. These agents do not know the reward *function* or the rewards (travel times) of other agents; they just know their own travel times. However, their decisions do influence the reward each receives.

In this simple model for adaptive choice, the function to compute the reward R_t^i each agent i receives at time t is computed as in Eq. 1. The parameters n_M and n_S represent the number of agents selecting the main and secondary alternative, respectively. B is a balancing factor to prevent negative rewards; thus it changes with $(n_M + n_s)$. This formula was used in [14] for studying the decision making process in route decision experiments with human subjects. Selten's group performed his experiments with 18 human subjects during 200 rounds. B was then set to 30 meaning that *in the equilibrium*, 12 drivers should select M and 6 should select S. This way the reward of each would be 10 units.

$$R_t^i = \begin{cases} R_M = \frac{4\times B}{3} - \frac{(n_M+n_s)}{3} + (2\times n_M) & \text{if } i \text{ selects } M \\ R_S = \frac{4\times B}{3} - \frac{2\times(n_M+n_S)}{3} + (3\times n_S) & \text{if } i \text{ selects } S \end{cases} \quad (1)$$

Based on the rewards received, agents then update a so-called "choice heuristic". Practically, it is the probability p_i according to which an agent i selects the main alternative (M). For instance, if it is 1 then the agent always selects M; if it is 0 it always selects S; otherwise the agent selects M with probability p_i.

In the adaptive scenario the agent i updates this heuristic with a given frequency. The update is based on the rewards i has obtained selecting M up to that point. The update of the heuristic considers the fraction of reward for a given alternative (e.g., R_M) divided by the sum of rewards provided by all alternatives, as in Eq. 2.

$$p_i = \frac{\sum_t R_M^i}{\sum_t R_M^i + \sum_t R_S^i} \quad (2)$$

Here, R_M^i is the reward agent i has accumulated (up to step t) when selecting M, while R_S^i denotes his success on alternative S. There is a feedback loop - the more an agent selects an alternative, the more information (in form of reward) it gets about it. Therefore an important factor is how often and in which intervals the heuristic is updated. This is especially relevant because the reward depends on the other agents.

This reinforcement learning based approach is used by us for the sake of comparison. Here, following [10] we set the learning probability to 0.2. This means that, on average, in every fifth round the agents are adapting their heuristic. According to these authors, this results in a configuration where, on average, the agents learn the optimal heuristic of $\frac{2}{3}$ (i.e., they end up selecting M with probability $\frac{2}{3}$).

As mentioned, in the present paper we use RBN's to equip the agents with a decision-making framework. This is appropriate for binary (i.e., boolean) decision-making, which aim at considering inputs from other agents in this decision-making process. To do so, we modify the reinforcement learning method described by [10], replacing the rewarding mechanism: instead of rewarding past actions, here the rewards are associated with the possible boolean functions an agent may have. Hence, each node has random boolean functions and uses them to determinate whether to use M or S.

In our case, each agent ($i \in N$) is a kind of node in a random boolean network, which has K neighbors randomly assigned. Each agent thus has K entries in its boolean function, which then outputs either 0 or 1, depending on the values of these K inputs and on the function the agent is using. The inputs are the actions of the K neighbors in the last time step (in the initial time step, all agents start with a random action). We assume that if a boolean function returns 0, this means an agent selects S; otherwise the decision is to use M.

At each time step, all agents (nodes) make their decision simultaneously. This decision leads to the equilibrium if exactly two-thirds of the agents use M.

This way, at each time step the quality of every function is measured and its score (s) is updated. The score is initially the reward R_t^i obtained. Based on the performance of the functions, they evolve so that eventually a good set of functions is found. Finding a good function for each agent could in principle be used to lead the system to the equilibrium. Of course, one cannot be sure that a function that works well for one agent will also work for a different agent. That is because the outcome of the function depends on its inputs (in this case, the agent's neighbors).

Our approach for adaptation of the functions that are used at local level uses a so-called ε-*greedy*. Hereby, each node has a set of possible functions to choose from. This choice may be made at random or greedily. Here, at every τ time steps, the node has to choose one among these functions to be used. In $1 - \varepsilon$ times, the choice is greedy, i.e., the node chooses the function that has yielded the highest score so far. It is random ε times. The initial value is $\varepsilon = 1$ to allow exploration. Then, every τ time steps, the value of ε is decreased by a δ value (annealing).

Once a function is selected, the actions from other K agents (whether they have used M or S in the last time step) act as input to this function. The function prescribes an action to the agent and its reward is computed based on all action selections (as in Eq. 1). This reward is then used as score s of a function.

5. EXPERIMENTS AND ANALYSIS OF THE RESULTS

In this section, we investigate the use of a set of boolean functions that represent different strategies of the agents for the route choice scenario.

5.1 Settings

Each driver holds a set of 10 functions that are randomly assigned to it. Of course, these functions tend to select each route with probability 50%, on average. Therefore it is necessary to learn to use functions biased for selecting M.

Experiments were performed where the number of drivers (N) is 18 and 900. These values were chosen to make a comparison between the results found in this paper and those presented in [10].

The first parameter that is investigated is the number of neighbors of each driver (K). The experiments consider scenarios where $K \geq 2$ in order to identify how much information agents need to make decisions that lead to the equilibrium of the system.

The number of boolean functions each agent holds controls how many strategies the agents can consider during the simulation. It is important to realize that the number of possible functions is related to the parameter K: the total number of boolean functions is 2^{2^K}. Hence, for an RBN with $K = 2$, the total number of boolean functions is 16 and when $K = 3$ this number is 256. For these two situations, because each agent holds 10 functions, an agent in fact may hold up to 62.5% of all possible functions when $K = 2$. When $K = 3$, that ration drops to only 3.9%. For $K = 5$, the number of functions is over 4 billion. This of course has a huge impact in the exploration space, and it is a reason why we do not use $K > 5$.

The parameter τ refers to the frequency with which a chosen function is selected, i.e., at each τ steps agents select a new function among those it holds. A low value of τ yields a faster change of strategies. However, when this value is moderately high, agents can get a better assessment of the chosen function due to the fact that it is used for a longer time.

As mentioned, the selection of one among them is ε-greedy. The parameter δ controls how ε decreases during the simulation. This parameter is a real value between 0 and 1; it multiplies the current value of ε. This process is applied whenever an agent selects a strategy. At the beginning of the simulation, $\varepsilon = 1$, allowing any function to be chosen. When ε is multiplied by δ, this reduces the exploration rate of available strategies.

Table 5:

Parameter	Values
N	18, 900
K	2, 3, 4
B	30 ($N = 18$), 2100 ($N = 900$)
τ	1, 5, 10
δ	0.9, 0.95, 0.99

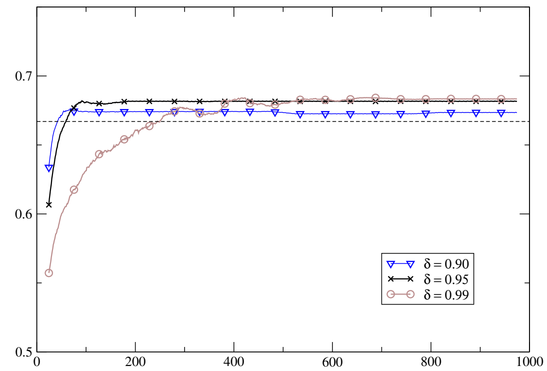


Figure 4: Fraction of drivers in M along time, for different values of δ; $N = 18$, $K = 2$ and $\tau = 1$ (dashed line indicates the desired equilibrium fraction).

For each set of parameters, 30 repetitions were performed and the plots presented in this section show the moving average of these simulations using a window of 50 time steps. Each simulation consists of 1000 time steps of route selection. Table 5 displays the parameters and their values as used in the experiments.

5.2 Results and analysis

Here we present and discuss the results of the simulations of route choice using the RBN formalism. Sections 5.2.1 and 5.2.2 consider the scenario where $N = 18$. The value of B is equal to 30. Parameter τ was set to 1, 5 and 10; δ is 0.9, 0.95 or 0.99. Smaller values for δ were not investigated because the value of ε would decrease too fast, resulting in a short exploration phase.

5.2.1 K = 2

Figure 4 shows the fraction of agents selecting M along time, for $K = 2$ and $\tau = 1$, changing the value of δ. We can see that the equilibrium is roughly missed, i.e., a number slightly above 12 drivers select M. This happens for any value of δ. This behavior is due to the fact that $\tau = 1$ and hence agents change function at every time step thus leading to an over reaction. Furthermore, it is possible to verify that the equilibrium is better approximated when $\delta = 0.9$. Here, because the 10 available functions (for each driver) cover 62% of the space of possible functions, the exploration phase does not need to last long. Also, for higher values of δ, the tendency is reached later because the exploration phase is longer.

The next experiments evaluate the same scenario where the parameter τ was set to different values. Figures 5 and 6 exhibit the result of the experiment for $\tau = 5$ and $\tau = 10$ respectively. One

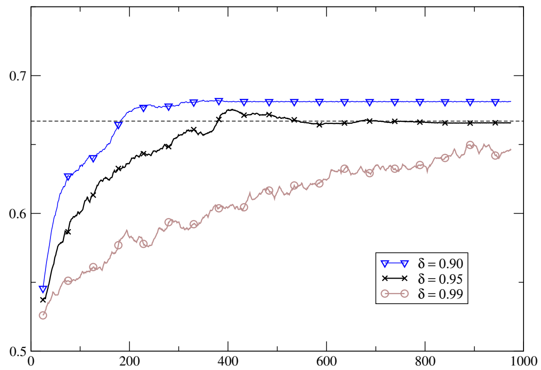


Figure 5: Fraction of drivers in M along time, for various values of δ; $N = 18$, $K = 2$ and $\tau = 5$.

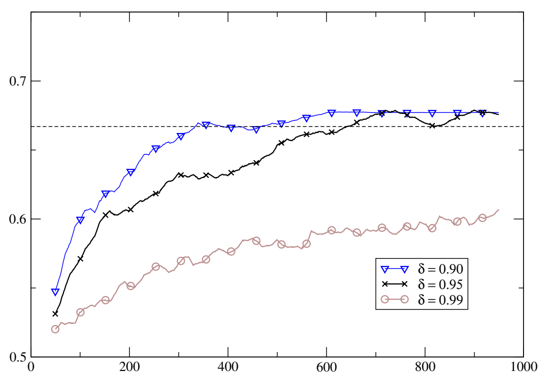


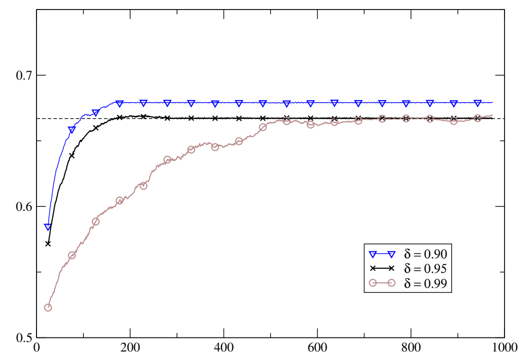
Figure 6: Fraction of drivers in M along time, for different values of δ; $N = 18$, $K = 2$ and $\tau = 10$.

notices a fluctuation in the values of the moving averages. This is due to the fact that agents keep the respective functions for a longer time. The higher τ, the higher the fluctuation. Also, it takes a longer time for agents to reach the equilibrium (which is, we recall, exactly 12 agents), and that this is reached sooner when $\delta = 0.95$. Contrarily to the previous case ($\tau = 1$), higher values of τ allow the selected functions to be better assessed since they are used for longer periods.

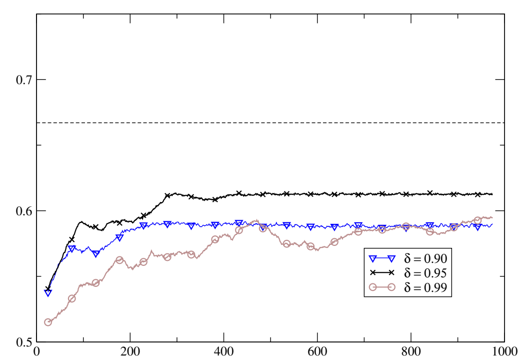
5.2.2 K > 2

The experiments described in the previous section were repeated for networks where $K = 3$, $K = 4$, and $K = 5$. In these cases, agents still have 10 functions to select from. It is important to recall that the number of possible functions grows rapidly with the increase of K. Thus, these experiments investigate if it is possible to reach an equilibrium still using that number of functions.

Figure 7 displays the results from simulations where $K = 3$ and $K = 4$. It is observed that when $K = 3$, it was possible to approximate the equilibrium when $\delta = 0.95$ and $\delta = 0.99$ (Figure



(a) $K = 3$



(b) $K = 4$

Figure 7: Fraction of drivers in M along time, for various values of δ and K: $K = 3$ (a) and $K = 4$ (b) ($\tau = 1$ and $N = 18$).

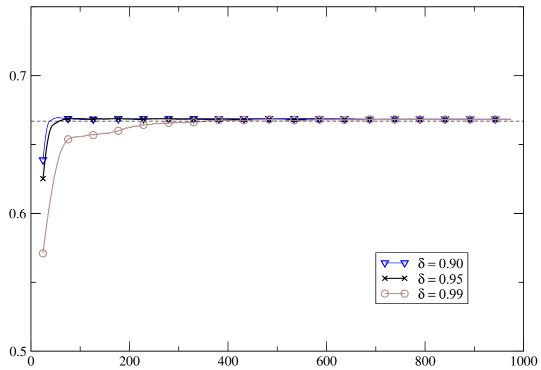


Figure 8: Fraction of drivers in M along time, for different values of δ; $N = 900$ and $K = 2$.

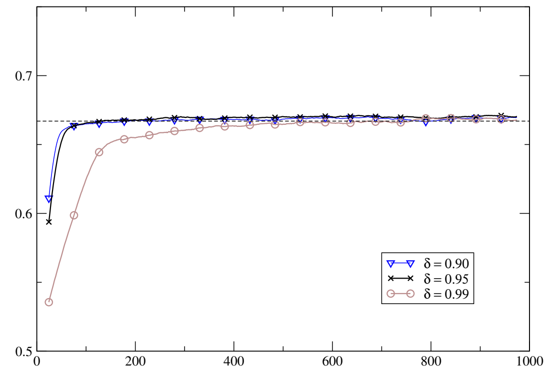


Figure 9: Fraction of drivers in M along time, for various values of δ; $N = 900$ and $K = 3$.

7 (a)). This result was not observed when $K = 4$, where the best result was a fraction of 0.613 of agents in M, when $\delta = 0.95$ (Figure 7 (b)). Although the plot is not shown here, we remark that when $K = 5$, as expected, drivers distribute among the two routes equally because the space of possible combination of functions is huge and agents are not able to make enough exploration during the simulation time. In fact, also the value of δ is not appropriated (it needs to be higher, approaching 1).

5.2.3 N = 900

In the experiments presented in previous sections, it was possible to notice that the equilibrium of the system has been achieved for networks where $N = 18$, for both $K = 2$ and $K = 3$. This section shows the experiments when $N = 900$. Again, this value was chosen to enable a comparison with [10]. Here B is set to 2100.

Figure 8 displays the simulation results when $K = 2$ and $\tau = 1$. The results show that the equilibrium was reached for all values of δ. The average fraction of agents in M at the end of the simulation is 0.668. The equilibrium is now exactly reached due to the higher number of agents. Also, notice that all plots for $N = 900$ depict less fluctuation, as expected, because the action of each single agent has less influence in the overall system.

The results obtained for $K = 3$ and $\tau = 1$ are very similar and can be seen in Figure 9.

Similarly to [10], we classify drivers into five categories according to the biases of the last boolean function selected by them:

- side route drivers ($0 \leq$ bias < 0.2),
- drivers with side route tendency ($0.2 \leq$ bias < 0.4),
- random drivers ($0.4 \leq$ bias < 0.6),
- drivers with main route tendency ($0.6 \leq$ bias < 0.8), and
- main route drivers ($0.8 \leq$ bias < 1).

Figure 10 shows the proportion of drivers in each category. The result differs from [10]: there is no concentration of main route drivers and side route drivers, that is, few agents have a high bias for one of the routes. We can see that most of the drivers have a bias for the main route.

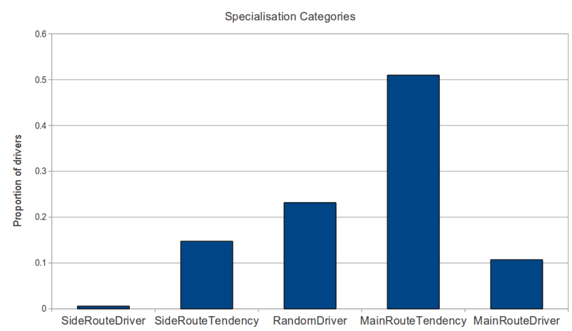


Figure 10: Classes of drivers at end of simulation ($N = 900$, $K = 3$, $\delta = 0.9$ and $\tau = 1$)).

6. CONCLUSIONS AND FUTURE WORK

This paper continues our long term work on simulation of route choice from the point of view of individual drivers. Contrarily to macroscopic approaches discussed in Section 2, we believe that the efficiency at system level, i.e., the equilibrium, can also be achieved by means of self-organization at individual level. To this aim we have investigated several techniques to evolve strategies that are grounded the mechanisms of self-organization.

In the present paper we use the well-known formalism introduced by Kauffman, the RBN's. This approach has the implication that agents are not considered isolated decision-makers (as, e.g., in [10] where the agent learned by observing its own rewards) but, rather, as a social network. The behavior of these agents is influenced by experiences exchanged within a group of acquaintances. Thus, we address not only the individual agents behavior, but also how information sharing could influence the emergence of a behavior among autonomous agents so that a coordinated situation emerges, which depicts efficiency at system level, out of individual decision-making. This, we remark, is the very basic objective of the collectives and self-organized systems.

We have performed simulations changing the main parameters of a RBN, namely N and K, as well as the parameters related to the evolution of the boolean functions. These are the frequency of function change, and the rate of exploration. Results show that with a relatively low number of boolean functions assigned to each driver, they are able to reach an equilibrium without using only their own experiences, which may be restricted. Also, it is not necessary that an agent models a high number of acquaintances for the approach to work. Rather, considering a lot of "friends" in the mapping input-action selection is likely to prevent a good coordination among these agents (as shown when we have increased K to 4 or 5).

This of course does not mean that one agent shall not have lots of acquaintances. On the contrary: we plan to investigate what happens if each agent is connected to a high number of agents, while being also able to select those connections that shall indeed contribute a good input in the boolean function. This decision about whether or not to include the input of some agent in one's network is to be based on the performance of that agent. Besides this extension, we also plan to have a network structured as a kind of scale-free network in which K varies from node to node. This will allow us to investigate the eventual importance of hubs in the network.

Acknowledgments

We thank the anonymous reviewers for their suggestions and for pointing out future research directions. Authors are partially supported by CNPq.

7. REFERENCES

[1] T. Arentze and H. Timmermans. Albatross: overview of the model, application and experiences. In *Innovation in Travel Modeling 2008 Conference, Portland, June 2008*, 2008.

[2] M. Balmer, K. Meister, M. Rieser, K. Nagel, and K. W. Axhausen. Agent-based simulation of travel demand: Structure and computational performance of MATSim-T. In *2nd TRB Conference on Innovations in Travel Modeling, Portland, June 2008*, 2008.

[3] A. L. C. Bazzan, R. H. Bordini, G. K. Andriotti, R. Viccari, and J. Wahle. Wayward agents in a commuting scenario (personalities in the minority game). In *Proc. of the Int. Conf. on Multi-Agent Systems (ICMAS)*, pages 55–62. IEEE Computer Science, July 2000.

[4] A. L. C. Bazzan, M. de Brito do Amarante, T. Sommer, and A. J. Benavides. ITSUMO: an agent-based simulator for ITS applications. In R. Rossetti, H. Liu, and S. Tang, editors, *Proc. of the 4th Workshop on Artificial Transportation Systems and Simulation*. IEEE, September 2010.

[5] A. L. C. Bazzan, M. Fehler, and F. Klügl. Learning to coordinate in a network of social drivers: The role of information. In K. Tuyls, P. J. Hoen, K. Verbeeck, and S. Sen, editors, *Proceedings of the International Workshop on Learning and Adaptation in MAS (LAMAS 2005)*, number 3898 in Lecture Notes in Artificial Intelligence, pages 115–128, 2006.

[6] A. L. C. Bazzan and F. Klügl. Case studies on the Braess paradox: simulating route recommendation and learning in abstract and microscopic models. *Transportation Research C*, 13(4):299—319, August 2005.

[7] T. Chmura and T. Pitz. An extended reinforcement algorithm for estimation of human behavior in congestion games. *Journal of Artificial Societies and Social Simulation*, 10(2), 2007.

[8] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J Theor Biol*, 22(3):437–467, March 1969.

[9] S. A. Kauffman. *The Origins of Order*. Oxford University Press, Oxford, 1993.

[10] F. Klügl and A. L. C. Bazzan. Route decision behaviour in a commuting scenario. *Journal of Artificial Societies and Social Simulation*, 7(1), 2004.

[11] F. Klügl and A. L. C. Bazzan. Simulated route decision behaviour: Simple heuristics and adaptation. In R. Selten and M. Schreckenberg, editors, *Human Behaviour and Traffic Networks*, pages 285–304. Springer, 2004.

[12] F. Klügl, A. L. C. Bazzan, and J. Wahle. Selection of information types based on personal utility - a testbed for traffic information markets. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 377–384, Melbourne, Australia, July 2003. ACM Press.

[13] S. Panwei and H. Dia. A fuzzy neural approach to modeling behavioural rules in agent-based route choice simulations. In A. L. C. Bazzan, B. Chaib-Draa, F. Klügl, and S. Ossowski, editors, *Proceedings of the 4th Workshop on Agents in Traffic and Transportation, AAMAS 2006, May, 9th, Hakodate, JP*, pages 70–78, 2006.

[14] R. Selten, M. Schreckenberg, T. Pitz, T. Chmura, and J. Wahle. Experimental investigation of day-to-day route choice-behaviour. In R. Selten and M. Schreckenberg, editors, *Human Behaviour and Traffic Networks*, pages 1–22. Springer-Verlag, 2004.

[15] K. Tumer and D. Wolpert. A survey of collectives. In K. Tumer and D. Wolpert, editors, *Collectives and the Design of Complex Systems*, pages 1–42. Springer, 2004.

[16] J. Wahle, A. L. C. Bazzan, and F. Klügl. The impact of real time information in a two route scenario using agent based simulation. *Transportation Research Part C: Emerging Technologies*, 10(5–6):73–91, 2002.

[17] J. G. Wardrop. Some theoretical aspects of road traffic research. In *Proceedings of the Institute of Civil Engineers*, volume 2, pages 325–378, 1952.

[18] T. Yamashita, K. Izumi, and K. Kurumatani. Analysis of the effect of route information sharing on reduction of traffic congestion. In F. Klügl, A. L. C. Bazzan, and S. Ossowski, editors, *Application of Agent Technology in Traffic and Transportation*, pages 99–112. Birkhäuser, 2004.

[19] T. Yamashita and K. Kurumatani. New approach to smooth traffic flow with route information sharing. In A. L. C. Bazzan and F. Klügl, editors, *Multi-Agent Systems for Traffic and Transportation*. IGI Global, 2009.