TRAFFIC SIMULATION AT JUNCTION: NON-NORMATIVE PRACTICES VS. DEADLOCK

Arnaud Doniec[1], Stéphane Espié[2], René Mandiau[3], Sylvain Piechowiak[4]

ABSTRACT

Simulating traffic at junction is quite complex. The common approach simplifies drastically the process: the traffic "solver" lets vehicles enter into the junctions only when their trajectories are not in conflict. This solution is sometimes sufficient, but is not acceptable when the question is to mimic actual behaviour. In particular these common approaches don't take the non-normative practices usually performed by real driver into account. Our aim is to improve traffic simulation models for driver's behaviour at junctions by using a multi-agent approach. The framework is the INRETS-MSIS ARCHISIM behavioural traffic simulation model, which is based on psychological fundings. In this model, actors of the traffic system interact between themselves in order to produce traffic phenomena. Each simulated driver has its own knowledge, goal and strategies. In crossroad situation for example, they decide to stop or go according to their own assessment of their relative priorities. This fully distributed algorithm gives good results, but leads in some cases to deadlocks (particularly for situations with many left-turn manoeuvres). A second attempt was to improve this algorithm by taking account of anticipation mechanism: simulated drivers recognize the situation and enter into the junction if and only if they anticipate not to create a deadlock situation. The use of this algorithm removes nearly all the deadlocks while "mimicking" actual behaviour. We will first present the problematic of traffic in junctions. We will then present the framework of our work. We will discuss the game theory based algorithm used for the simulation of the go / no go process, and the anticipation mechanism we have developed to ensure a mitigation of deadlocks. We will conclude by explaining our prospects and the possible use of such a simulation.

KEY WORDS

Multi-agent system, coordination, anticipation, traffic simulation.

---

[1] Phd Student, MSIS Department, INRETS, 2 avenue du Général Malleret-Joinville, 94114 Arcueil, France, Phone +33 1 47 40 70 03, FAX +33 1 45 47 56 06, doniec@inrets.fr

[2] Director, MSIS Department, INRETS, 2 avenue du Général Malleret-Joinville, 94114 Arcueil, France, Phone +33 1 47 40 70 23, FAX +33 1 45 47 56 06, espie@inrets.fr

[3] Professor, LAMIH, University of Valenciennes, Le Mont Houy, 59313 Valenciennes, France, Phone +33 3 27 51 14 38, FAX +33 3 27 51 13 16, mandiau@univ-valenciennes.fr

[4] Professor, LAMIH, University of Valenciennes, Le Mont Houy, 59313 Valenciennes, France, Phone +33 3 27 51 14 38, FAX +33 3 27 51 13 16, piechowiak@univ-valenciennes.fr

INTRODUCTION

Traffic simulation at junction is a complex problem. The common approach consists in using a centralized scheduler over the traffic flow on each branch of the crossroad. This approach constitutes a really drastic simplification of the problem since the scheduling process lets enter into the intersection only the vehicles whose trajectories are not in conflict. This solution is sometimes sufficient, but is not acceptable when the question is to mimic actual behaviour. It allows neither a good accuracy of traffic flow (and jam...) for dense situations, nor a "realistic" traffic usable for a driving simulator.

The MSIS team from INRETS develops for many years a driving simulator dedicated to the studies of driver behaviours and traffic. This approach is original since it combines the use of a behavioural traffic simulation model (ARCHISIM) (Espié 1995) and a simulator (SIM²). In the ARCHISIM model, the traffic is considered as an emergent phenomena which results from the actions and interactions of the different actors of the simulation: car drivers, motorcyclists, road operators, pedestrians…

The computing model of ARCHISIM follows the multi-agent principles (Demazeau 2001). Each simulated driver is considered as an autonomous software agent with its own knowledge, goals and strategies. It evolves into a virtual environment made up of roads and intersections. Many agents may want to reach a same part of the road at the same time: this conflict can lead to an accident or a deadlock. To avoid such a situation, the agents have to coordinate with each other as in real life.

TRAFFIC SIMULATION AT JUNCTIONS

OVERVIEW

Crossing an intersection is a driving task which differs from one country to the next. For example, for the latin drivers, this task is mostly "competitive". Conversely, in the northern country, the crossing is less competitive and more "cooperative". Therefore, traffic simulation at junction requires a coordination between conflictual traffic flows (intersection and merging streams) in a high dynamic environment. This problematic can be expressed as a multi-agent coordination issue. First works on multi-agent assumed that all agents act in order to achieve a common goal: distributed planning (von Martial 1992), task sharing (Gasser et al 1988), distributed research algorithm (Yokoo 2001, Doniec et al 2005). This type of coordination can be qualified as cooperative. On the opposite, the competitive coordination tries to give an answer to the problem in which many agents try to achieve their own goal in respect of such properties of the global system. This field of research is more recent and few works exist (Shoham et al 1995, Aknine et al 2000).

In (Champion et al 2003), the authors present a competitive coordination mechanism dedicated to the traffic simulation at junction. This mechanism which is fully distributed and integrated in the ARCHISIM model is described in the next section.

MODELING TRAFFIC SITUATION AT JUNCTION

Each complex crossroad can be viewed as a succession of simple junctions and a traffic situation can be expressed as a combination of elementary situations. An elementary situation is made up of two vehicles *x* and *y* sharing a priority relation. The figure 1 describes the four elementary situations which can occur at a simple junction.

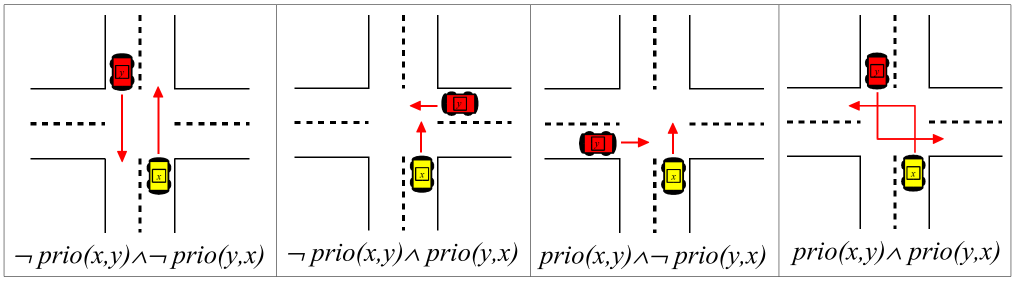

Figure 1: Elementary traffic situations at junction

Each situation is characterized by potential conflict points over the mobiles' trajectories. In (Champion et al, 2003), the authors introduce a modeling of these situations based on game theory. Each agent can choose between two possible actions *Go* and *Stop* which allow to calculate the longitudinal acceleration of the simulated vehicle. The situation is therefore described as a payoff matrix which associates earning or penality to the four possible strategies: *x Go / y Go , x Stop / y Go , x Go / y Stop , x Stop / y Stop* (Figure 2).

$\begin{matrix} & Go & Stop & \\ & \begin{pmatrix} (a_1, a_1) & (a_3, 0) \\ (0,0) & (0,0) \end{pmatrix} & & \begin{matrix} Go \\ Stop \end{matrix} \end{matrix}$	$\begin{matrix} & Go & Stop & \\ & \begin{pmatrix} (-b_2, -b_1) & (b_6, 0) \\ (0, b_3) & (0,0) \end{pmatrix} & & \begin{matrix} Go \\ Stop \end{matrix} \end{matrix}$	$\begin{matrix} & Go & Stop & \\ & \begin{pmatrix} (-b_1, -b_2) & (b_3, 0) \\ (0, b_6) & (0,0) \end{pmatrix} & & \begin{matrix} Go \\ Stop \end{matrix} \end{matrix}$	$\begin{matrix} & Go & Stop & \\ & \begin{pmatrix} (-c_1, -c_1) & (c_3, 0) \\ (0, c_3) & (0,0) \end{pmatrix} & & \begin{matrix} Go \\ Stop \end{matrix} \end{matrix}$
¬ prio(x,y)∧¬ prio(y,x)	*¬ prio(x,y)∧ prio(y,x)*	*prio(x,y)∧¬ prio(y,x)*	*prio(x,y)∧ prio(y,x)*
$\{a_1, a_3, b_1, b_2, b_3, b_6, c_1, c_3\} \in \mathbb{N}^{*+}$			

Figure 2: Elementary matrix

An aggregation of elementary matrix allows to describe complex traffic situations evolving more than two agents. The choice of the best strategy is performed by a payoff maximisation whose details can be found in (Champion, 2003).

Before executing this algorithm each simulated driver has to determine with which agents it has to coordinate. This selection of agents issue is crucial for the conflicts resolution between the agents in the intersection and consequently for the realism of the simulated traffic.

AGENT SELECTION ISSUE

In our context, the agent'behaviour can be defined by the way it is going to perceive the agents which it is in conflict with. As we have to simulate the behaviour of real drivers it looks interesting to use non normative behaviour. Indeed, the highway code is more or less respected. Sometimes the non-respect of the code is necessary. For example, in the case of a double left turn, the rule is that the vehicles drive round each other. In the case of two trucks, this rule leads to a deadlock. Using non normative behaviour makes easier the getting of a larger disparity between the simulated drivers and thus the emergence of realistic traffic situations.

CONFLICT POINTS AND PRIORITY RELATIONS

For each pair of agents x and y in crossroad situation, there is a conflict point for which it is possible to define two distances:

- *distConflict(x,y)*, the distance between x and the conflict point
- *distConflict(y,x)*, the distance between y and the conflict point

When the vehicles are on different roads, the conflict point is situated at the intersection of their trajectories. When the vehicles are on the same road, two cases are possible according to their direction. If both x and y go straight on, no conflict point exists and we can set: $distConflict(x,y)=+\infty$ and $distConflict(x,y)=+\infty$. On the contrary, if x and y are in a double left turn situation, two conflict points are conceivable. To remove this ambiguity, we can discriminate the conflict point by a time-to-conflict calculation depending on speed and acceleration of the two mobiles.

In reality, a driver can not possibly try to manage his conflicts with more than 4 or 5 other vehicles. Consequently, in ARCHISIM, each agent tries to perform coordination with no more than 4 other mobiles. The selection process of agents for coordination is applied in two steps. The first one consists in searching both the conflict points p_i $(1 \leq i \leq 4)$ and the associated mobiles. During the second step, the agent x determines for each p_i the mobile y_{pi} the most constraining from the point of view of the priority relation between x and y_{pi}. When two agents y'_{pi} and y_{pi} share an equivalent relation, the choice is performed according to the distance to the conflict point: $min(distConflict(y'_{pi},x), distConflict(y_{pi},x))$.
The selection of agents issue depends on the priority relation perceived by an agent x. A simple way to introduce non normative behaviour is to consider this relation as an aggregation of different types of priorities.

PRIORITY AGGREGATION

We have implemented and tried several types of aggregation. We present in the following section the one which offers the best realism and we refer to it by *agg3*. This aggregation *agg3* is described in the figure 3 by a set of rules expressed as implication. The predicate *prioCode* represents the normative priority of highway code whereas *prioPhysical* stands for the priority relating to an awkward entry in the conflict zone. We also introduce the predicates *impatience* and *moveOffAgain*. The first one expresses the impatience state of the agent and the second indicates if the agent accelerates after having stopped.

$$(1)\quad prioPhysical(x,y) \rightarrow prio_A(x,y) \wedge \neg prio_A(y,x)$$
$$(2)\quad prioCode(x,y) \wedge prioCode(y,x) \wedge turnBehind(y,x) \rightarrow \neg prio_A(x,y) \wedge prio_A(y,x)$$
$$(3)\quad prioCode(x,y) \wedge prioCode(y,x) \wedge \neg turnBehind(x,y) \rightarrow prio_A(x,y) \wedge \neg prio_A(y,x)$$
$$(4)\quad prioCode(x,y) \wedge \neg prioCode(y,x) \wedge impatience(x) \rightarrow prio_A(x,y) \wedge \neg prio_A(y,x)$$
$$(5)\quad prioCode(x,y) \wedge \neg prioCode(y,x) \wedge \neg impatience(x) \wedge (distConflict(x,y) < distConflict(y,x)) \rightarrow prio_A(x,y) \wedge \neg prio_A(y,x)$$
$$(6)\quad prioCode(x,y) \wedge \neg prioCode(y,x) \wedge \neg impatience(x) \wedge (distConflict(x,y) \geqslant distConflict(y,x)) \wedge moveOffAgain(y) \rightarrow prio_A(x,y) \wedge \neg prio_A(y,x)$$
$$(7)\quad prioCode(x,y) \wedge \neg prioCode(y,x) \wedge \neg impatience(x) \wedge (distConflit(x,y) \geqslant distConflit(y,x)) \wedge \neg moveOffAgain(y) \rightarrow \neg prio_A(x,y) \wedge prio_A(y,x)$$
$$(8)\quad \neg prioCode(x,y) \wedge prioCode(y,x) \wedge (moveOffAgain(x) \vee impatience(x)) \wedge (distConflict(x,y) < distConflict(x,y)) \rightarrow prio_A(x,y) \wedge \neg prio_A(y,x)$$
$$(9)\quad \neg prioCode(x,y) \wedge prioCode(y,x) \wedge (moveOffAgain(x) \vee impatience(x)) \wedge (distConflict(x,y) \geqslant distConflict(x,y)) \rightarrow \neg prio_A(x,y) \wedge prio_A(y,x)$$
$$(10)\quad \neg prioCode(x,y) \wedge prioCode(y,x) \wedge \neg moveOffAgain(x) \wedge \neg impatience(x) \rightarrow \neg prio_A(x,y) \wedge prio_A(y,x)$$
$$(11)\quad \neg prioCode(x,y) \wedge \neg prioCode(y,x) \rightarrow \neg prio_A(x,y) \wedge \neg prio_A(y,x)$$

Figure 3: Priorities aggregation

Without being explicitely expressed, this aggregation takes the priority relative to the speed into account. For example, a mobile arriving in a crossroad with a high speed will be able to consider that it has priority. Conversely, a mobile will be able to consider it has no longer priority over a faster vehicle. When the speeds are near from zero, distances are considered instead of speed (rules 5,6,7,8,9). On the other hand, when speeds are significant, all boolean tests using *distConflict* are remplaced by time to conflict, gap acceptance, braking time.

Non-normative behaviour and deadlocks

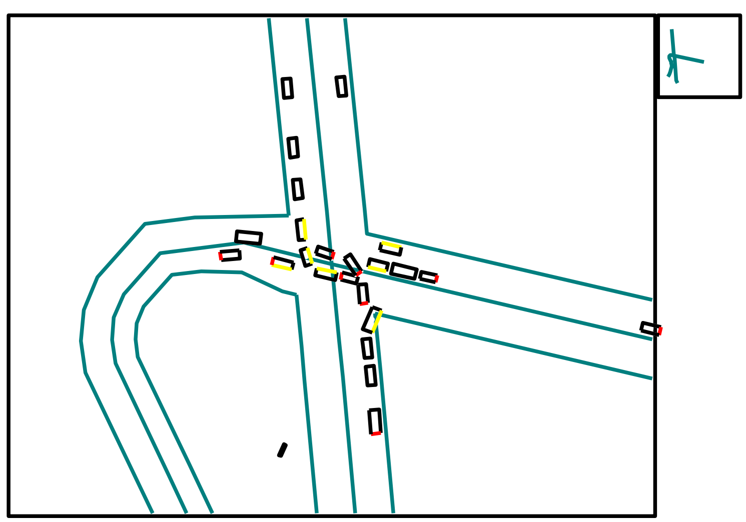

Figure 4: Example of deadlocks in an intersection

The aggregation presented in the previous section allows to obtain non normative practices and consequently realistic traffic situations. However, such practices can lead to deadlocks inside the intersection. When simulating high density of vehicles, agents can store themselves in the inner center of the crossroad and consequently create a deadlock. An example of such a situation is presented on figure 4.

In real life, even if drivers perform individual and non-normative practices, deadlocks situations are quite uncommon since drivers anticipate. From the psychological standpoint, and based on in-deepth studies conducted in actual situations, Saad research work demonstrates the importance of the anticipation when driving (Saad, 1992): “driving is anticipating”. Taking this aspect of the driving task into account, we propose an action selection model based on anticipation.

ANTICIPATION OF DEADLOCKS SITUATIONS

Anticipation is a general concept. First works about anticipation have been led by biologists and psychologists who tried to explain the adaptative behaviour of some animals. In 1985, Robert Rosen proposed for the first time a general framework of anticipation. The definition he introduced makes the link between knowledge of the future and decision making at present time: “An anticipatory system is a system containing a predictive model of itself and / or of its environment that allows it to change current state at an instant in accord with the model predictions pertaining to a later instant”(Rosen, 1985).

In the field of multi-agent system, anticipation is used in various applications: creation of complex and adaptative behaviour in video games (Laird, 2001), planification in high dynamic environment (Marc, 2004), eliminating of non cooperative situations (Georgé, 2004). In this section, the idea is to use anticipation in order to remove, before the coordination, the actions which can induce deadlocks.

MODEL OF ANTICPATION

The model we introduce consists in partitioning the environment representation of an agent in two parts: desired and undesired states. Deadlocks are of course considered as undesired states. With this background, the anticipation process consists in building a projection of the environment in the future and testing the existence of undesired states. The major difficulties is to describe and represent the environment.
The architecture of ARCHISIM allows agents to get, at each step of the simulation, a symbolic description of their environment with information about other perceived vehicles: position on the road, current speed, acceleration, etc. We enrich this symbolic vision by considering three relations of livelocks between two vehicles:

- $bph_z(x,y) \equiv x$ is physically blocked by y from the point of view of agent z
- $bpha_z(x,y) \equiv z$ perceives that x will be physically blocked by y
- $bphr_z(x,y) \equiv y$ has priority over x from the point of view of agent z

A conjunction set of these relations allows us to describe easily the contextual traffic situation occuring in a crossroad. The computation of these relations does not violate the rules of distribution of data since they can be infered by real drivers.

Inference of undesired states

To be able to infer undesired states from livelocks relations previously introduced, we use a based on constraints formalism which describes a set of variables taking their value in domains and constrained by binary relations. Each agent in crossroad situation builds a representation consisted of vehicles present inside the intersection (variables) and having livelock relation between each other (constraints).

Each vehicle acts according to a temporal domain that can be reduced by the livelock relations. For example, $dom(x) = [1,+\infty[$ expresses that x will be able to move during the interval $t+1$ to $+\infty$ but $dom(x)=[1,4]\cup[8,10]$ expresses that x will be blocked between $t+5$ and $t+7$. The reduction of domain can be computed by using constraints propagation algorithm.

General anticipation algorithm

```
function Anticipe(ListeActions LA, ReseauDeContraintes CN)
begin
  propage(CN);                                  (i1)
  for each A in LA do
    copieDeCN <- CN;                            (i3)
    LC <- determineEffetsDirects(A);            (i4)
    ajouteConstraintes(LC,CN);                  (i5)
    propage(CN);                                (i7)
    if existeUnEtatNonDesires(CN)               (i8)
      then
        efface(A,LA)                            (i9)
    end;
   CN <- copieDeCN;                             (i10)
  end;
 return LA;                                     (i11)
end
```

Figure 5: Anticipation algorithm

The first step of the algorithm (figure 5) consists in performing a first propagation over the constraint networks `CN` previously built by the agent *x*. This allows to reduce the domain of each variable. Then for each possible action of the agent *x*, the algorithm determines the effects of it. These effects entail the addition or the eliminating of livelocks relations. When the effects of an action are estimated, the algorithm makes an update of the constraint network and performs a new propagation. Then starts the research of undesired states. For example, a nil domain for the agent *x* constitutes a future deadlock situation. If an action will create such a situation, it is removed from the list of action `LA`.

The complexity of our algorithm depends on methods used for the propagation (Bessiere et al, 1995), (Mackworth, 1977) which complexity is bounded by $O(ed^2)$ and $O(ed^3)$ (d is the max length of the considered domains and e the total number of relations in the network). As this algorithm is run at each time step by each of the n agents involved in the situation, the global complexity is in the worst case $O(ned^2)$.

This theorical complexity must be considered in context. Indeed, by keeping a length of domain quite short, the main term d^3 remains fair. Moreover, we must consider the fact that all vehicles inside the crossroad will not execute the algorithm: the vehicles which are already blocked do not need to anticipate since they can not move anymore.

EVALUATION

A first type of evaluation has been led by using scenarii test over crossroad without signalisation and giving way to traffic coming from the right (Doniec, 2006). Since our initial goal is to reduce the number of deadlocks induced by the use of non normative behaviour, we also proceed by comparison between different simulations realised with a same traffic demand on the same road network.

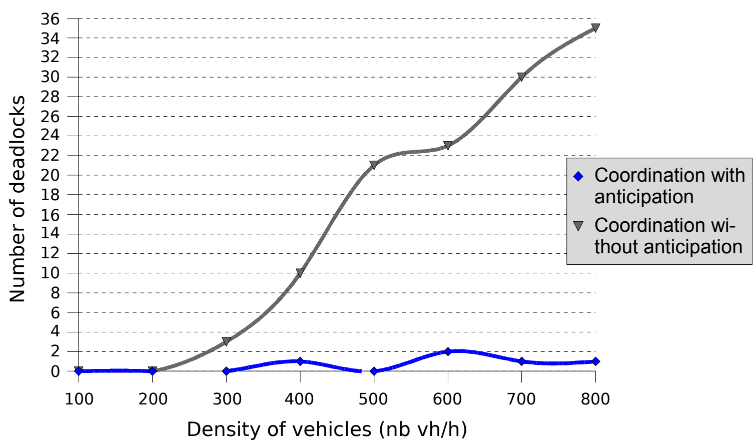

Figure 6: Reduction of the number of deadlocks with anticipation

The figure 6 presents the results of such a simulation. The first curve is obtained only with the coordination mechanism and the aggregation *agg3* presented in section 3.2. It expresses the number of deadlocks occuring during the simulations in terms of the density (number of vehicles per hour) simulated. We can notice that the number of deadlocks grows quickly. The second curve is obtained by adding the anticipation mechanism to the coordination. It shows a notable reduction of the number of deadlocks.

CONCLUSIONS

In this article we have exposed a traffic simulation at junction issue. Compared to actual traffic simulation tools, we try to ensure a realistic traffic inside the intersection. To do this we use non normative behaviour in the behavioural traffic simulation model: ARCHISIM.

When we want to simulate a high density of traffic, the use of non normative practices induces livelocks and deadlocks. In order to reduce deadlocks situations, we have introduced an anticpation mechanism which allows agents to eliminate their actions which can create deadlocks in the future. The evaluations over unit test and scenarii are really satisfactory. The next step of our work consist in evaluating the impact of non normative behaviour and anticipation over statistical result of traffic simulation.

REFERENCES

S. Espié, (1995) "ARCHISIM : Multi-actor parallel architecture for traffic simulation". *Proceedings of the Second World Congress on Intelligent Transport Systems*.

Y. Demazeau (2001), VOYELLES. Habilitation à diriger des recherches, Université de Grenoble.

F. von Martial, (1992). Coordination by negociation based on a connection of dialogue states with actions. DAI Workshop.

L. Gasser (L.) et A. Bond, (1988) Readings in distributed artificial intelligence. Morgan Kaufman, San Mateo.

M. Yokoo, (2001) Distributed Constraint Satisfaction : Foundations of Cooperation in Multi-agent Systems, Springer.

A. Doniec, S. Piechowiak, R. Mandiau, (2005) A DisCSP Solving Algorithm Based on Sessions. *Proceedings of FLAIRS.*

Y. Shoham, M. Tennenholtz, (1995) Computational Social Systems : offline design, *Journal of Artificial Intelligence Research*.

S. Aknine, S. Pinson, M.F. Shakun, (2000) Méthodes de coordination multi-agent par formation de coalition, *Systèmes Multi-agents, Ingénierie, Technologie, Expérimentations, Hermes.*

A. Champion, S. Espié, R. Mandiau, C. Kolski, (2003) A game-based multi-agent mechanism – application to road traffic and driving simulations. *Summer Computer Simulation Conference*.

A. Champion, (2003) Mécanisme de coordination multi-agent fondé sur les jeux : application à la simulation comportementale de traffic routier en situation de carrefour. Thèse de doctorat, Université de Valenciennes.

F. Saad (1992). In-depth analysis of interactions between drivers and the road environment : contribution of on-board observations and subsequent verbal report - *In Proceedings of the 4th Workshop of ICTCT, University of Lund, Bulletin 110.*

R. Rosen, (1985) Anticipatory systems – Philosophical, Mathematical, and Methodological Foundations, Pergamon Press.

J. Laird, (2001). It knows what you're going to do : adding anticipation to a quakebot. *Proceeding of Autonomous Agent*.

F. Marc, A. El Fallah-Seghrouchni, I. Degirmenciyan-Cartault, (2004) Modelling control and validation of multi-agent plans in highly dynamic context. *Proceeding of AAMAS*.

J. Georgé, (2004) Résolution de problèmes par émergence : étude d'un environnement de programmation émergente. Thèse de doctorat, Université de Toulouse.

A. Mackworth *(1977)*, Consistency in networks of relations, Artificial Intelligence.

C. Bessière, Freuder E., Regin J. (1995) Using inference to reduce arc consistency computation. *Proceeding of the International Conference on Artificial Intelligence*.

A. Doniec, R. Mandiau, S. Piechowiak and S. Espié (2006) L’anticipation comme modèle d’interaction : application à la coordination multi-agent en simulation. *Actes RFIA* , Tours.