HAL
open science

Diagnostic décentralisé à l'aide d'automates cellulaires

Nicolas Gauville, Nazim Fatès, Irène Marcovici

To cite this version:

Nicolas Gauville, Nazim Fatès, Irène Marcovici. Diagnostic décentralisé à l'aide d'automates cellulaires. JFSMA 2019 - 27emes Journées Francophones sur les Systèmes Multi-Agents, Institut de Recherche en informatique de Toulouse et l'Association française pour l'Intelligence Artificielle, Jul 2019, Toulouse, France. pp.96-105. hal-02195799

HAL Id: hal-02195799

https://inria.hal.science/hal-02195799v1

Submitted on 26 Jul 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Diagnostic décentralisé à l'aide d'automates cellulaires

Nicolas Gauville[a,c] nicolas.gauville@loria.fr
Nazim Fatès[a] nazim.fates@inria.fr
Irène Marcovici[a, b] irene.marcovici@univ-lorraine.fr

[a] Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

[b] Université de Lorraine, CNRS, Inria, IECL, F-54000 Nancy, France

[c] Safran Electronics & Defense

Résumé

Nous nous intéressons au problème du diagnostic de défaillances dans un réseau distribué. Lorsque les composants du réseau sont susceptibles de tomber en panne, comment détecter le moment où le taux de composants défaillants dépasse un certain seuil sans faire appel à une autorité centrale ? Notre objectif est d'avoir une estimation de l'état général du réseau par le seul biais d'interactions locales des composants avec leurs voisins. En particulier, nous souhaitons qu'un consensus émerge sous forme d'état d'alerte lorsque le taux de défaillance dépasse un certain seuil. Nous utilisons le modèle des automates cellulaires pour proposer des solutions dans le cas d'un réseau ayant une structure de grille. Nous comparons trois méthodes d'auto-organisation du réseau, en partie inspirées de phénomènes physiques ou biologiques. Comme domaine d'application, nous avons en vue les réseaux de capteurs ou tout système fonctionnant de manière décentralisée.

Mots-clés : *Émergence, auto-organisation, viabilité ; résolution collective de problèmes ; déploiement de SMA, résistance aux pannes, fiabilité*

Abstract

We address the problem of detecting failures in a distributed network. If some components can break down over time, how can we detect that the failure rate has exceeded a given threshold without any central authority ? Our aim is to have an estimate of the global state of the network, only through local interactions of components with their neighbours. In particular, we wish to reach a consensus on an alert state when the failure rate exceeds a given threshold. We use the model of cellular automata in order to propose solutions in the case of a network with a grid structure. We compare three methods of self-organisation that are partly inspired by physical and biological phenomena. As an application, we envision sensor networks or any type of decentralised system.

Keywords: *Emergence, self-organisation, viability ; collective resolution of problems ; deployment of MAS, reliability to failures, fiability*

1 Diagnostic décentralisé

Nous nous intéressons ici à un réseau d'éléments tous identiques en interaction avec leurs voisins immédiats. Nous supposons que les éléments du réseau ont un état normal de fonctionnement et un état défaillant, lequel survient de manière aléatoire et irréversible. Notre objectif est de détecter de manière décentralisée le moment où le taux d'éléments défaillants dépasse un seuil critique fixé à l'avance. L'idée est de provoquer un changement de comportement global brutal lorsque ce seuil d'éléments défaillants est atteint, soit en diffusant un état d'alerte spécifique, soit en créant un *consensus global* sur l'état des éléments.

Afin de traiter ce problème dans un cadre mathématique simple, nous explorons le cas où le réseau d'éléments en interaction peut être représenté par un automate cellulaire. Le réseau est donc formé par un ensemble d'agents immobiles, les *cellules*, qui ont un état discret et sont disposées sur une grille bidimensionnelle finie. Les cellules changent leur état à chaque pas de temps en fonction d'une loi, appelée *règle de transition locale*, qui est uniforme (les cellules sont toutes régies selon la même loi). La localité de la loi exprime le fait que chaque cellule met à jour son état en fonction de son propre état et de celui des ses voisines immédiates.

Le cadre formel que nous utilisons rejoint celui des systèmes multi-agents réactifs, où les agents prennent des décisions en disposant de très peu de mémoire [11]. Ici, la structure de grille que nous avons choisie contraint fortement les interactions entre agents. Néanmoins, comme nous le

verrons par la suite, nous avons sélectionné les règles que nous présentons pour leur robustesse à des modifications de topologies qui rendraient le réseau plus irrégulier. De plus, ce cadre très simple permet de faciliter la reproductibilité des expériences. Nous avons effectué les simulations à l'aide du logiciel FiatLux [1], mais le modèle peut être reproduit très facilement par d'autres moyens.

La conception d'un système qui réalise de manière *décentralisée et robuste* un diagnostic pour dire si un taux critique de cellules défaillantes a été franchi à un moment donné est un problème difficile. En effet, nous travaillons dans un cadre qui conjugue trois contraintes fortes : a) localité des échanges d'information, b) uniformité de la règle de transition, c) petit nombre d'états pour les cellules. Ces trois contraintes jointes imposent donc de ne pas calculer *directement* le ratio du nombre de cellules défaillantes sur le nombre de cellules total. En effet, un tel calcul imposerait une centralisation, totale ou partielle, qui contredirait les contraintes du problème et serait en désaccord avec notre objectif, qui est d'avoir un fonctionnement *robuste* du système : si l'information était centralisée dans des noeuds particuliers, la défaillance (ou l'attaque) de tels noeuds pourrait mettre en danger l'ensemble du système.

En somme, le défi est de concevoir un système qui puisse se diagnostiquer lui-même tout en subissant une détérioration : les cellules défaillantes ne peuvent changer d'état ou transmettre d'information. On peut comparer ce problème à d'autres tâches comme la classification de la densité [4], pour laquelle il s'agit de déterminer quel est l'état majoritaire sur une grille de cellules possédant un état binaire, en respectant les mêmes contraintes (localité, uniformité de la règle de transition, petit nombre d'états). En effet, il s'agit dans les deux cas d'obtenir de manière décentralisée un consensus sur une propriété globale du système. Ici, pour ne pas être trop éloignés des cas d'application réels, nous souhaitons également que notre système soit robuste à l'asynchronisme des mises à jour des composants et qu'il ne soit pas spécifique à une forme particulière du réseau. Ce type d'approche a jusqu'ici été très peu considéré ; un point de vue similaire a cependant été adopté par Bénézit, qui a proposé différents modèles d'obtention de consensus sur des réseaux où circule une information [1]. D'autres pistes de recherche se développent également en vue de proposer des algorithmes totalement décentralisés, par exemple pour la maximisation de la couverture des réseaux de capteurs [12].

Dans cette optique, pensons au cas concret suivant : on dissémine un ensemble de capteurs sur un grand territoire. Ceux-ci ont une durée de fonctionnement limitée du fait de l'épuisement de leur batterie. On souhaite pouvoir détecter de manière totalement décentralisée, c'est-à-dire par un consensus, le moment où une fraction donnée de ces capteurs ne fonctionne plus, en vue de savoir quand il sera nécessaire d'intervenir pour ajouter de nouveaux capteurs qui pourront les remplacer. Notre système de diagnostic vise à répondre à ce genre de problème.

Dans la partie 2, nous présentons le formalisme utilisé pour décrire nos modèles. Nous étudions alors le fonctionnement de ces modèles et évaluons leur adéquation au problème du diagnostic décentralisé (parties 3, 4 et 5). Enfin, dans la partie 6 nous établissons une synthèse rapide de nos observations et discutons des perspectives de ce travail.

2 Automates cellulaires

De manière informelle, on peut définir un automate cellulaire comme un système composé d'une grille d'automates tous identiques, les cellules, ayant les caractéristiques suivantes :

1. Chaque cellule possède un état choisi parmi un ensemble fini d'états ; cet ensemble est noté Q.
2. À chaque itération, une cellule met à jour son état en fonction de son état courant et de celui de son voisinage, à l'aide d'une règle de transition locale.
3. Toutes les cellules appliquent la même règle de transition locale à chaque itération.

Un automate cellulaire est un cas particulier de système dynamique discret. Nous nous restreignons ici au cas des grilles bidimensionnelles. L'espace des cellules est défini par $\mathcal{L} = \{1, \ldots, n\} \times \{1, \ldots, m\}$ pour une grille de taille (n, m) (avec $n, m \in \mathbb{N}^*$).

Une configuration est un élément de $\mathcal{C} = Q^{\mathcal{L}}$, qui représente l'état de toutes les cellules de la grille à un moment donné.

Le voisinage $V(c)$ d'une cellule $c \in \mathcal{L}$ est un sous-ensemble fini de la grille, qui représente l'ensemble des cellules dont l'état est

1. http://fiatlux.loria.fr/

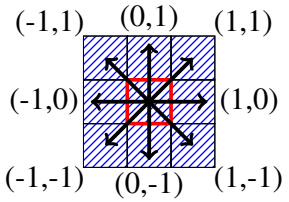

FIGURE 1 – Voisinage de Moore (cellule considérée entourée en rouge, voisines hachurées en bleu).

pris en compte par la règle de transition locale de l'automate cellulaire. Dans cet article, nous utiliserons le voisinage de *Moore*, avec la convention que la cellule centrale n'est pas incluse dans le voisinage (cf. figure 1) : on pose $V_M = \{-1, 0, 1\}^2 \setminus \{(0, 0)\}$ et le voisinage $V(c)$ de la cellule $c \in \mathcal{L}$ est défini par $V(c) = \{c + (i, j) : (i, j) \in V_{\mathrm{M}}\} \cap \mathcal{L}$. Avec cette définition, les cellules du bord ont un voisinage de taille plus petite. Les fonctions locales que nous introduisons sont bien définies pour les cellules du bord, puisqu'elles dépendent seulement des nombres de voisines dans chaque état.

Règle de transition locale f

Dans cette étude, nous nous restreignons au cas où la règle de mise à jour des cellules est de la forme $f : Q \times \mathbb{N}^Q \longrightarrow Q$: la fonction de transition locale f prend en entrée l'*état courant de la cellule* considérée ainsi que le *nombre de cellules voisines se trouvant dans chacun des états de* Q. Les règles de cette forme sont dites *totalisantes-externes* (*outer-totalistic*).

L'application de cette règle de transition locale se fait en utilisant une fonction intermédiaire qui compte le nombre de cellules du voisinage qui sont dans un état donné :

$$\begin{array}{rccl} \eta : & \mathcal{C} \times \mathcal{L} & \longrightarrow & \mathbb{N}^Q \\ & (x, c) & \longmapsto & (|\{c' \in V(c); x_{c'} = q\}|)_{q \in Q}. \end{array}$$

Fonction de transition globale ϕ

Le système dynamique final est donc obtenu à l'aide de la fonction de transition globale $\phi : \mathcal{C} \longrightarrow \mathcal{C}$, qui associe une configuration de $\mathcal{C}$ à chaque configuration, après application de la règle de transition locale à toutes les cellules. Elle est définie par :

$$\begin{array}{rccl} \phi : & \mathcal{C} & \longrightarrow & \mathcal{C} \\ & x & \longmapsto & y \text{ tel que } \forall c \in \mathcal{L}, y_c = f(x_c, \eta(x, c)). \end{array}$$

Nous noterons x^0 la configuration initiale (correspondant au temps zéro), puis pour $t \geqslant 0$, $x^{t+1} = \phi(x^t)$.

3 Règle de *Greenberg-Hastings*

Notre point de départ est de chercher une règle locale simple dont la propriété principale soit de faire vivre une « onde » sur le réseau qui se fige si le nombre de cellules défaillantes est trop important (voir figure 3). De telles propriétés ont déjà été observées dans une version probabiliste de la règle de Greenberg-Hastings : ce modèle permet de modéliser simplement la propagation de vagues excitatrices à travers une grille de cellules (voir ref. [2] et les références ci-incluses).

L'ensemble des états est défini par $Q = \{\mathtt{N}, \mathtt{E}, \mathtt{R}\}$, où $\mathtt{N}$ représente l'état *neutre*, $\mathtt{E}$ l'état *excité*, et $\mathtt{R}$ l'état *réfractaire*. L'évolution est régie par une règle locale f qui prend en argument l'état de la cellule et le nombre $n_{\mathtt{E}}$ de cellules voisines dans l'état excité, et est définie de la manière suivante : une cellule neutre qui possède au moins un voisin dans l'état excité devient excitée avec probabilité p_τ, elle reste neutre sinon. Une cellule excitée devient toujours réfractaire et une cellule réfractaire devient toujours neutre. Cette règle est formalisée figure 2. Notons que les cellules défaillantes ne changent plus d'état et n'interviennent pas dans les transitions des autres états.

Il est connu que les propriétés de diffusion de cette onde dépendent fortement de la valeur de la probabilité de transmission p_τ : en dessous-d'un seuil critique, les ondes disparaissent brutalement [10, 2, 9]. C'est cette propriété dite de *transition de phase* qui nous permet ici de déclencher l'alerte : le changement brutal de comportement induit un changement qui nous permet de réaliser notre diagnostic décentralisé.

3.1 Mécanisme d'alerte

La problème central qui se pose à nous est donc de pouvoir distinguer, de manière locale, les deux phases du système, c'est-à-dire la présence ou l'absence d'une onde formée d'état excités sur la grille.

En utilisant l'automate cellulaire de Greenberg-Hastings, notre approche consiste à introduire un automate cellulaire ayant la propriété suivante : les cellules changent d'état d'autant moins souvent qu'il y a de cellules défaillantes. Plus précisément, s'il y a peu de cellules défaillantes, les cellules restent *actives*, ce qui signifie que leur état change fréquemment lorsque la règle locale s'applique. À l'inverse, la présence de cellules défaillantes dans une zone donnée de la grille

$$\textbf{État N :}\quad \textbf{Si } n_{\mathtt{E}} \geqslant 1,\quad alors\quad f(\mathtt{N}, n_{\mathtt{E}}) = \begin{cases} \mathtt{E} \text{ avec probabilité } p_\tau \\ \mathtt{N} \text{ avec probabilité } 1 - p_\tau \end{cases}$$
$$\textbf{si } n_{\mathtt{E}} = 0,\quad alors\quad f(\mathtt{N}, n_{\mathtt{E}}) = \mathtt{N}$$
$$\textbf{État E :}\quad f(\mathtt{E}, n_{\mathtt{E}}) = \mathtt{R}$$
$$\textbf{État R :}\quad f(\mathtt{R}, n_{\mathtt{E}}) = \mathtt{N}$$

FIGURE 2 – Règle de Greenberg-Hastings

aura tendance à faire baisser l'activité des cellules. Nous utilisons donc un compteur d'inactivité, qui permet de faire en sorte qu'une cellule restée inactive durant un trop grand nombre de mises à jour se place dans un état spécifique d'alerte. Cet état se propage alors à son tour de proche en proche sur l'ensemble de la grille.

Cette méthode possède l'avantage de fonctionner en mode « tout ou rien » : l'état d'alerte est totalement absent de la grille tant qu'il n'a pas été produit en un lieu donné, puis il se propage en uniformisant les états des cellules.

Pour préciser les choses, notons $x_{i,j}^t$ l'état de la cellule $(i, j) \in \mathcal{L}$ au temps t. Les valeurs du compteur sont comprises entre 0 et une valeur maximale $T_{\max}$. À chaque pas de temps, lorsqu'une cellule change d'état, son compteur est remis à 0, tandis que si elle ne change pas d'état, son compteur augmente. Lorsque la valeur maximale $T_{\max}$ est dépassée, le compteur de la cellule prend un état de signalement spécial noté $\mathtt{S}$. Cet état $\mathtt{S}$ se maintient jusqu'à un éventuel changement d'état de $x_{i,j}^t$. Si une cellule a au moins quatre voisines dans l'état de signalement $\mathtt{S}$, elle passe dans l'état d'alerte $\mathtt{A}$, qui se propage ensuite sur la grille.

L'ajout de cet état intermédiaire de signalement $\mathtt{S}$ permet de s'assurer qu'il y a une zone contenant plusieurs cellules inactives contiguës avant de déclencher une alerte, ce qui permet d'être moins sensible à une anomalie localisée.

Considérons une cellule $(i, j) \in \mathcal{L}$, et notons $n_{\mathtt{A}}^t(i, j)$ et $n_{\mathtt{S}}^t(i, j)$ le nombre de cellules voisines de la cellule (i, j) qui sont au temps t dans l'état d'alerte $\mathtt{A}$ et de signalement $\mathtt{S}$, respectivement. L'état $\tau_{i,j}^{t+1}$ du compteur de la cellule (i, j) au temps $t + 1$ est alors défini de manière formelle dans la figure 4.

3.2 Test du modèle

Nous commencerons par nous intéresser au cas de l'apparition progressive des défaillances au cours du temps, en mesurant l'évolution des différentes cellules. Ensuite, nous nous intéresserons au cas statique où les grilles sont initialisées avec un taux fixe de cellules défaillantes. Dans ces expériences, nous mesurerons la probabilité $p_{\mathtt{A}}$ qu'une grille initialisée avec une densité de cellules défaillantes donnée se place en état d'alerte après un certain nombre de pas de temps.

Dans un premier temps, nous nous plaçons donc dans le cas où les défaillances apparaissent de manière aléatoire uniforme : à chaque pas de temps, chaque cellule a une probabilité $p_{\text{def}} = 5.10^{-5}$ de devenir défaillante. La figure 5 présente l'évolution de la densité de cellules en état d'alerte et la densité de cellules défaillantes au cours du temps. Les densités des états d'alerte sont relatives au nombre de cellules non défaillantes.

On constate que la densité relative de cellules en état d'alerte passe brutalement de 0 à 1, du fait de la propagation de l'état d'alerte. Lorsqu'on reproduit l'expérience, on constate que la proportion de cellules défaillantes présentes au moment où ce saut se produit varie faiblement. Ce modèle répond donc au problème posé dans la mesure ou nous atteignons bien un état d'alerte qui se présente sous forme d'un consensus général sur l'état des cellules.

Examinons maintenant comment le seuil varie en fonction du paramètre du modèle p_τ. Pour cela, nous nous plaçons désormais dans un cas statique où les défaillances sont fixées une fois pour toutes. Initialement chaque cellule peut être défaillante avec probabilité $d_{\mathtt{D}}$, ou non défaillante auquel cas son état initial est tiré avec équiprobabilité parmi les états $\mathtt{E}$, $\mathtt{N}$, $\mathtt{R}$. Pour estimer la position du seuil en fonction du paramètre du modèle, nous effectuons, pour une valeur de $d_{\mathtt{D}}$ fixée, Z tests, dans lesquels nous faisons évoluer ces configurations initiales pendant T pas de temps. Nous notons ${p_{\mathtt{A}}}^T(d_{\mathtt{D}})$, la proportion des Z échantillons pour lesquels au bout de T pas de temps, la densité relative de cellules en état d'alerte dépasse 95%. Ce seuil peut être choisi arbitrairement proche de 100% mais nous avons souhaité éviter cette valeur limite de façon à ce

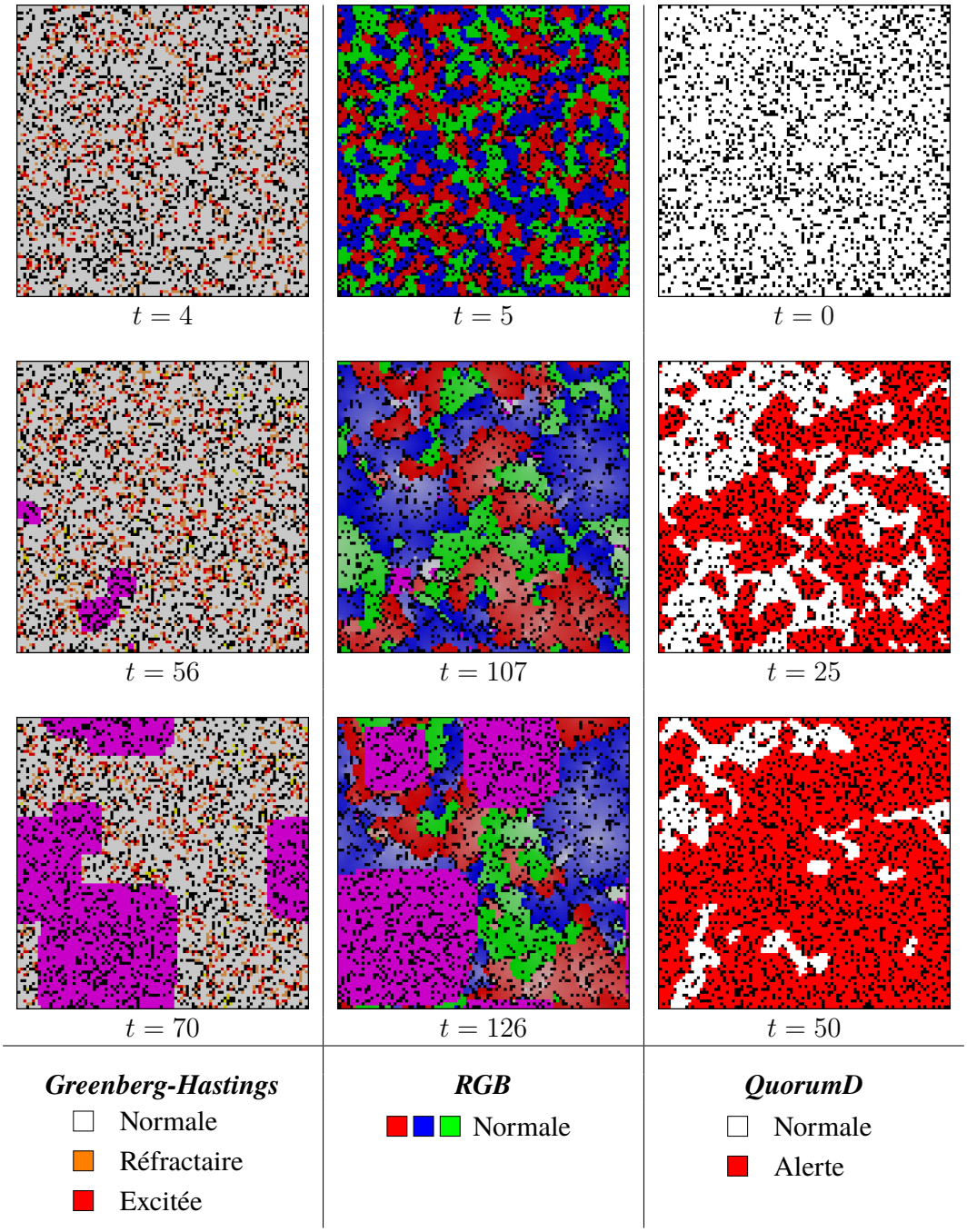

FIGURE 3 – Représentation de configurations typiques pour les trois modèles proposés. Les cellules défaillantes sont en noir. Sur les modèles Greenberg-Hastings et RGB, l'état d'alerte qui se propage est représenté en violet, l'état de signalement en jaune ; sur le modèle *QuorumD*, l'état d'alerte est en rouge.

que la présence de quelques cellules isolées restées à l'état normal ne perturbe pas la mesure.

Les résultats de l'expérience présentés figure 9 nous montrent que nous pouvons ajuster le seuil de cellules défaillantes détecté avec le paramètre p_τ.

3.3 Cas de l'asynchronisme

Nous avons supposé jusqu'ici que la règle locale s'appliquait de manière synchrone, c'est-à-dire, que toutes les cellules étaient mises à jour simultanément. Cependant, nous souhaitons que notre

Si	$n^t_{\mathtt{A}}(i,j) \geq 1$ ou $\tau^t_{i,j} = \mathtt{A}$,	**alors**	$\tau^{t+1}_{i,j} = \mathtt{A}$	(propagation de l'état d'alerte),
sinon si	$n^t_{\mathtt{S}}(i,j) \geq 4$,	**alors**	$\tau^{t+1}_{i,j} = \mathtt{A}$	(apparition de l'état d'alerte),
sinon si	$x^{t+1}_{i,j} \neq x^t_{i,j}$,	**alors**	$\tau^{t+1}_{i,j} = 0$,	
sinon si	$x^{t+1}_{i,j} = x^t_{i,j}$,	**alors**	$\tau^{t+1}_{i,j} = \begin{cases} \tau^t_{i,j} + 1 & \text{si } \tau^t_{i,j} < T_{\max} \\ \mathtt{S} & \text{sinon.} \end{cases}$	

FIGURE 4 – Règle de propagation de l'alerte

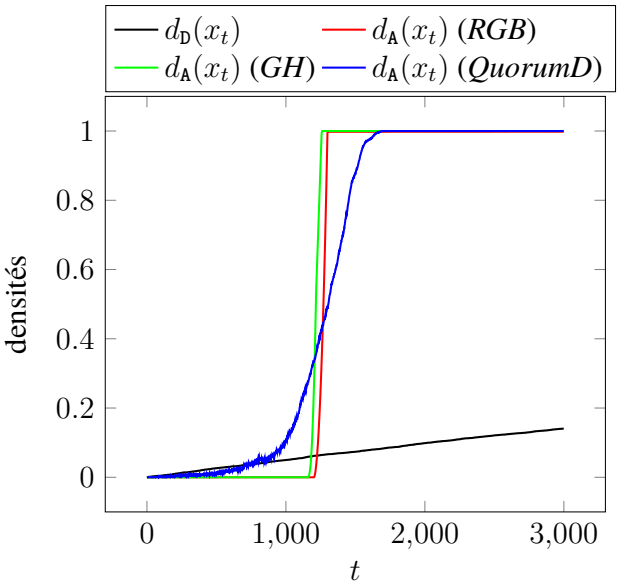

FIGURE 5 – Évolution des densités de cellules défaillantes et en alerte pour les modèles *RGB*, *Greenberg-Hastings* ($p_\tau = 0,31$) et *QuorumD* ($\lambda = 5$), sur des grilles de 100×100 cellules. Les paramètres des trois modèles ont été choisis pour aligner les seuils de cellules défaillantes détectés. Les trois modèles passent en état d'alerte après un certain nombre d'itérations.

méthode de diagnostic soit robuste à l'asynchronisme pour permettre au système de fonctionner en l'absence d'une horloge centralisée. En effet, dans le cas d'applications pratiques comme des réseaux de capteurs, il est préférable de ne pas supposer une synchronisation parfaite des transitions [3].

Notre prochaine expérience a donc pour but d'évaluer la robustesse du modèle à l'asynchronisme. Pour ce faire, nous appliquons une méthode de mise à jour dite α-asynchrone [6] : lors de chaque itération de la fonction de transition globale, chaque cellule est mise à jour avec une probabilité α, et laissée dans son état avec probabilité $1 - \alpha$.

La figure 10 reproduit l'expérience précédente (avec une valeur fixe des paramètres) en faisant varier le taux de synchronisme α. On constate que pour ce modèle, le seuil se déplace fortement. Par conséquent, si la façon dont les éléments sont mis à jour dans le système varie au cours du temps, il se produira aussi un déplacement de la valeur du seuil de détection. Ce modèle n'est donc pas robuste à l'asynchronisme. Nous allons donc chercher une règle robuste à l'asynchronisme.

4 Modèle *RGB*

Comme le modèle de *Greenberg-Hastings*, le modèle *RGB* que nous allons présenter a la propriété d'adopter deux comportements radicalement différents en fonction du taux de cellules défaillantes présentes sur la grille.

4.1 Présentation du modèle

Nous introduisons maintenant une nouvelle règle, également à trois états, mais pour laquelle la fonction de transition locale présente davantage de symétrie par rapport à ces trois états. Il s'agit d'un nouveau modèle, qui peut cependant rappeler la famille des automates cycliques, pour lesquels la règle locale fait changer les états des cellules en suivant un ordre pré-déterminé [7, 8].

La règle utilise trois états : $Q = \{R, G, B\}$. Soit q l'état courant d'une cellule, on note q^+ l'état suivant dans le cycle R-G-B (cf. figure 7), n_{q^+} le nombre de voisins dans cet état, et $n_{\mathtt{D}}$ le nombre de voisines défaillantes. La règle locale consiste à passer à l'état q^+ de manière certaine si on a au moins trois voisines dans l'état q^+, et avec une probabilité p_{tr} si on a seulement deux voisines dans l'état suivant. L'état courant est conservé dans les autres cas. La règle est donnée formellement dans la figure 6. À nouveau, les cellules défaillantes ne peuvent changer d'état et n'interviennent pas dans les transitions des autres états.

$$
\begin{array}{llrcl}
 & & f: & Q \times \mathbb{N}^3 & \longrightarrow & Q \\
\textbf{Si} & n_{q^+} \geqslant 3, & f(q; n_R, n_G, n_B) & = & q^+ \\
\textbf{si} & n_{q+} = 2, & f(q; n_R, n_G, n_B) & = & \begin{cases} q^+ \text{ avec probabilité } p_{tr} \\ q \text{ avec probabilité } 1 - p_{tr} \end{cases} \\
\textbf{si} & n_{q+} \leqslant 1, & f(q; n_R, n_G, n_B) & = & q.
\end{array}
$$

FIGURE 6 – Règle *RGB*

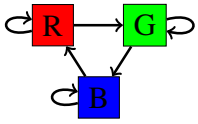

FIGURE 7 – Transitions possibles de l'automate RGB.

4.2 Expériences

De même que pour le modèle de *Greenberg-Hastings*, le modèle *RGB* voit son activité diminuer lorsque le nombre de cellules défaillantes augmente, ce qui permet de déclencher l'alerte au bout d'un certain temps (voir figure 5). Nous reprenons le même mécanisme que précédemment pour déclencher l'alerte, on peut donc à nouveau détecter un seuil de cellules défaillantes trop important. Ici, c'est p_{tr} qui va nous servir à régler le moment où l'alerte se déclenche. La figure 9 reprend le même protocole expérimental que pour *Greenberg-Hastings* (état initial équiprobable, même temps de mesure, même nombre d'échantillons) et donne des résultats similaires.

Qu'en est-il de la robustesse à l'asynchronisme ? Nous constatons sur les résultats présentés figure 10 que cette fois, le seuil détecté ne varie plus en fonction du taux de synchronisme α. Nous avons donc un net avantage à choisir cette règle dans un contexte où le mode de mise à jour des cellules n'est pas connu à l'avance.

4.3 Problème du passage à l'échelle

La contrepartie de ce fonctionnement en mode binaire est ce que nous appelons *le problème du passage à l'échelle* : si une petite partie de la grille reste inactive suffisamment longtemps pour déclencher l'état d'alerte, et ce même si le taux de cellules défaillantes à détecter n'est pas encore atteint, l'alerte sera déclenchée partout. Ceci peut se produire notamment dans le cas où des cellules défaillantes apparaissent très proches les unes des autres et coupent une zone de cellules du reste de la grille. La probabilité que ce cas de figure arrive est d'autant plus grande que la taille de la grille est grande, ce qui interdit donc un passage à l'échelle de la méthode. Nous proposerons donc d'autres solutions ne présentant pas ce problème.

5 Méthode *QuorumD*

Nous cherchons ici à trouver une solution dont les propriétés de déclenchement d'alerte soient indépendantes de la taille de la grille. Pour cela, nous abandonnons le mécanisme précédent de propagation d'alerte et nous considérons un état d'alerte qui peut apparaître et disparaître localement.

En dessous du seuil critique, l'état d'alerte est présent de manière partielle ; au-dessus du seuil critique, un *consensus global* s'établit et l'état d'alerte envahit totalement la grille.

5.1 Présentation du modèle

Idéalement, nous souhaitons obtenir une règle de transition possédant les caractéristiques suivantes :

- Une cellule dont toutes les voisines sont en état d'alerte ou défaillantes passe en état d'alerte (*c1*).
- Une cellule dont toutes les voisines sont en état normal passe à l'état normal (*c2*).
- Dans le cas contraire, la cellule passe en état d'alerte avec une probabilité d'autant plus grande que le nombre de cellules voisines en état d'alerte ou défaillantes est grand (*c3*).

Nous avons essayé différentes formes de fonctions pour définir la probabilité de passer à l'état d'alerte. Nous avons constaté que des règles reposant sur des combinaisons linéaires du nombre de cellules dans chaque état ne permettent pas d'obtenir de comportement satisfaisant. La règle que nous présentons ici est inspirée des modèles utilisés en physique statistique et en biologie. Par analogie avec le phénomène de détection de quorum (*quorum sensing*) présent dans les populations de bactéries, nous l'appelons *QuorumD*. L'obtention de l'état d'alerte est réglé par un paramètre λ qui permet de faire varier le taux de

$$
\begin{array}{llll}
f: & Q \times \mathbb{N}^Q & \longrightarrow & Q \\
\textbf{Si } n_{\mathtt{N}} = 0, & f(q; n_{\mathtt{A}}, n_{\mathtt{N}}, n_{\mathtt{D}}) & = & \mathtt{A} \qquad (c1) \\
\textbf{Si } n_{\mathtt{A}} = n_{\mathtt{D}} = 0, & f(q; n_{\mathtt{A}}, n_{\mathtt{N}}, n_{\mathtt{D}}) & = & \mathtt{N} \qquad (c2) \\
\textbf{Sinon,} & f(q; n_{\mathtt{A}}, n_{\mathtt{N}}, n_{\mathtt{D}}) & = & \begin{cases} \mathtt{N} \text{ avec probabilité } \frac{e^{\lambda \frac{n_{\mathtt{N}}}{S}}}{e^{\lambda \frac{n_{\mathtt{N}}}{S}} + e^{\lambda \frac{n_{\mathtt{A}}+n_{\mathtt{D}}}{S}}} \\ \mathtt{A} \text{ avec probabilité } \frac{e^{\lambda \frac{n_{\mathtt{A}}+n_{\mathtt{D}}}{S}}}{e^{\lambda \frac{n_{\mathtt{N}}}{S}} + e^{\lambda \frac{n_{\mathtt{A}}+n_{\mathtt{D}}}{S}}} \end{cases} \qquad (c3) \\
 & & & \text{avec } S = n_{\mathtt{N}} + n_{\mathtt{A}} + n_{\mathtt{D}}.
\end{array}
$$

FIGURE 8 – Règle sans propagation (*QuorumD*).

cellules défaillantes à détecter. Une version formelle de cette règle est donnée sur la figure 8.

5.2 Test du modèle

La figure 5 présente l'évolution des densités de cellules défaillantes et de cellules en état d'alerte au cours du temps. Nous constatons que l'état d'alerte commence par coexister avec l'état normal en gardant une densité faible, puis lorsque le taux de cellules défaillantes devient trop grand, il prend le dessus pour envahir la grille. On peut donc parler ici d'un *comportement émergent*, au sens où l'état global d'alerte résulte ici d'une prise de décision véritablement décentralisée. Ceci rend la règle non seulement robuste à un changement de la taille de la grille, mais aussi à une concentration atypique de défauts à un endroit de la grille.

Nous souhaitons maintenant savoir dans quelle mesure nous pouvons fixer le seuil de cellules défaillantes à détecter en changeant la valeur du paramètre λ. La figure 9 nous montre que ce paramètre permet bien de régler le seuil de cellules défaillantes à détecter. De plus, le passage d'un état sans alerte à un état d'alerte se fait de manière plus précise en fonction du nombre de cellules défaillantes.

5.3 Résistance à l'asynchronisme

Comme pour les autres modèles, nous faisons varier le taux de synchronisme. La figure 10 présente la variation du déclenchement de l'alerte en fonction de ce taux : on observe une excellente robustesse du système à ces variations. Ceci est particulièrement encourageant dans la perspective d'utiliser ce modèle dans des situations concrètes.

5.4 Robustesse au changement d'échelle

Rappelons que nous cherchons à trouver une règle qui permette de détecter un seuil de défaillance fixé, même lorsque la taille du réseau varie. Par exemple, dans le cas des réseaux de capteurs, il arrive que de nouveaux capteurs soient ajoutés au réseau après le déploiement initial de celui-ci.

La figure 11 montre que les modèles de *Greenberg-Hastings* et *RGB* ont tendance à déclencher l'alerte plus tôt pour une taille de grille plus grande. En effet, plus la grille est grande et plus la probabilité qu'une zone ou les défaillances sont localement nombreuses provoque l'alerte en dessous du seuil est grande. En revanche, le modèle *QuorumD* a l'avantage d'avoir un comportement indépendant de la taille de la grille : les mesures de $p_{\mathtt{A}}$ ne varient pas lors du passage à l'échelle, ce qui montre que le passage dans l'état d'alerte généralisé résulte bien d'un consensus décentralisé.

6 Synthèse

Nous avons présenté une méthode de diagnostic décentralisé à l'aide d'automates cellulaires ayant des règles de transition simples et utilisant peu d'états. Ceci nous permet d'envisager des applications concrètes où cet automate cellulaire fonctionne en parallèle du système à diagnostiquer, tout en étant économe en quantité de calcul et en mémoire. Nous avons exploré deux approches différentes : la diffusion d'un état d'alerte (modèles *Greenberg-Hastings* et *RGB*), et l'obtention d'un consensus décentralisé (modèle *QuorumD*). Les trois modèles présentés semblent être des solutions adéquates au problème du diagnostic décentralisé, ayant chacune un comportement différent en réponse à différentes perturbations. Les modèles de *Greenberg-*

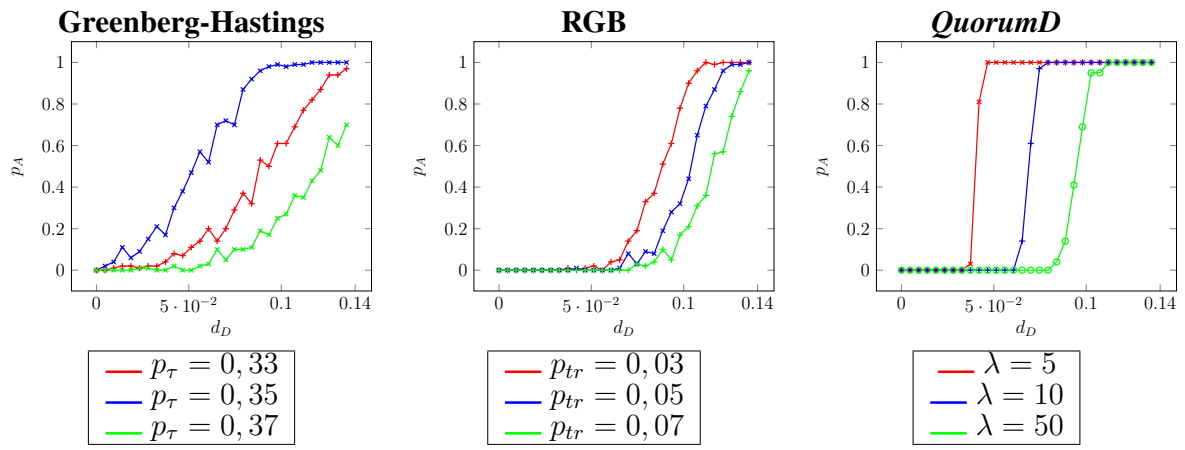

FIGURE 9 – Mesure de p_A pour différents paramètres de seuil d'alerte (mesures effectuées sur des grilles de 100×100 cellules sur une moyenne de $Z = 100$ tests).

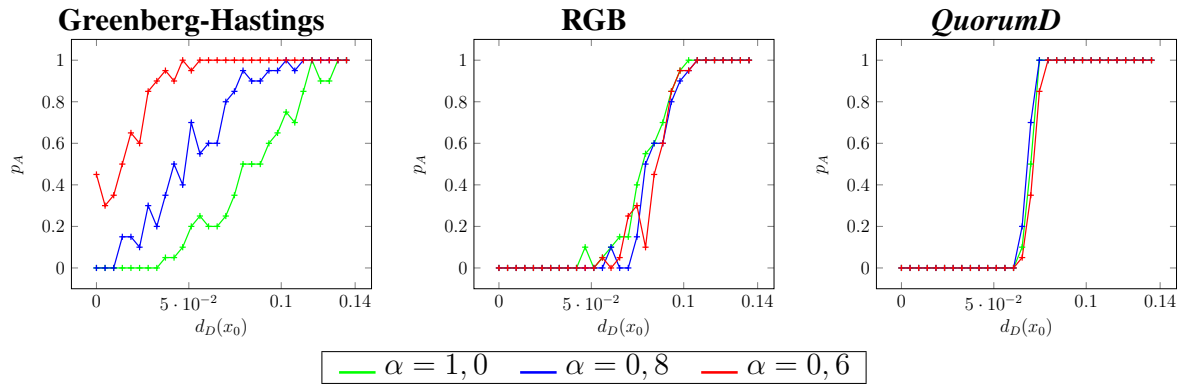

FIGURE 10 – Mesure de p_A pour différents taux de synchronisme (mesures effectuées sur des grilles de 100×100 cellules sur une moyenne sur 20 tests, $\lambda = 10, p_\tau = 0,35, p_{tr} = 0,02$).

Hastings et *RGB* ont l'avantage de ne pas faire apparaître d'état d'alerte en dessous du seuil à détecter, mais cette méthode interdit le passage à l'échelle direct (sans réajuster les paramètres). En revanche, même si le modèle *QuorumD* fait apparaître l'état d'alerte de manière plus progressive, il possède l'avantage d'être plus précis dans la détection du seuil de cellules défaillantes, d'être plus facile à paramétrer, et d'être plus robuste aux changements de mise à jour (asynchronisme). De plus, il est le seul des trois modèles à être robuste au changement d'échelle.

Le travail présenté d'ici est une première étude montrant la pertinence des automates cellulaires en vue de réaliser un diagnostic décentralisé. L'espace des règles étant gigantesque, notre travail d'exploration doit être poursuivi afin de mieux cerner les possibilités de solution au problème. Les expériences mériteraient également d'être approfondies, notamment en s'intéressant aux écart-types des valeurs obtenues.

Nous souhaitons désormais affiner notre compréhension des différents éléments entrant en compte dans la dynamique des automates cellulaires : l'influence du voisinage, du nombre d'états, de la forme des règles de transition, etc. En effet, il est difficile de prédire les différents comportements émergents observés et d'isoler les rapports entre les différents paramètres et le comportement global du système. Par ailleurs, d'autres types de défaillances sont envisageables et il serait intéressant de comparer l'effet de nos modèles d'auto-diagnostic distribué dans ces contextes.

Pour être en mesure d'appliquer ces méthodes sur le terrain, la prochaine étape sera d'examiner ces méthodes sur des topologies non régulières [5]. La forme des règles proposées ici a été choisie pour être aussi robuste que possible aux changements de topologie et des expériences préliminaires ont montré des résultats encourageants sur des topologies irrégulières [9]. En-

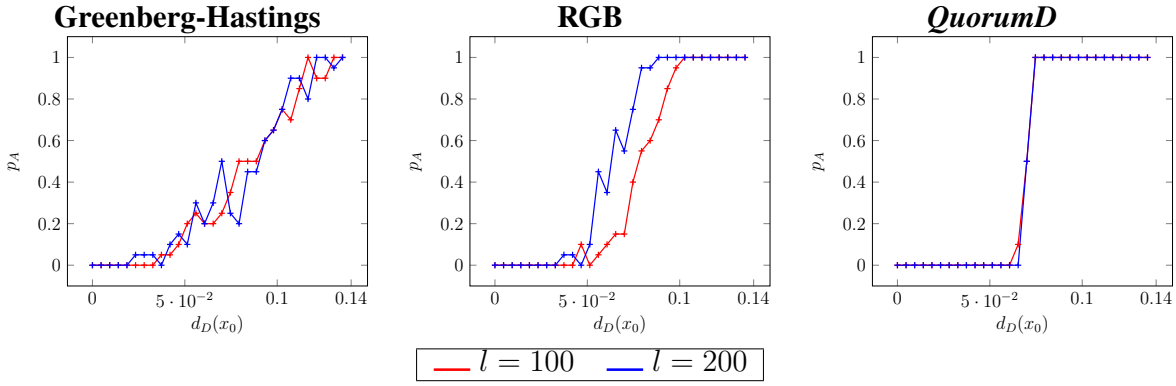

FIGURE 11 – Mesure de p_A pour différents taux de cellules défaillantes sur différentes tailles de grilles ($l \times l$) après 2000 itérations (moyenne sur 20 tests, $\lambda = 10, p_\tau = 0,35, p_{tr} = 0,02$).

fin, notons que les méthodes de transmission de l'information par automates cellulaires peuvent s'intégrer dans des systèmes multi-agents réactifs. On peut par exemple imaginer des situations où les agents tirent profit des informations de leur environnement pour localiser les zones où se trouvent le plus de défaillances et y engager des procédures de réparation.

Références

[1] Florence BÉNÉZIT, Patrick THIRAN et Martin VETTERLI : Interval consensus : From quantized gossip to voting. *In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'09*, pages 3661–3664, 2009.

[2] Hugues BERRY et Nazim FATÈS : Robustness of the critical behaviour in the stochastic Greenberg-Hastings cellular automatom model. *Int. Journ. of Unconventional Computing*, 7:65–85, 2011.

[3] Olivier BOURÉ : *Le simple est-il robuste ? une étude de la robustesse des systèmes complexes par les automates cellulaires*. Thèse de doctorat, Université de Lorraine, 2013. Université de Lorraine.

[4] Ana BUŠIĆ, Nazim FATÈS, Jean MAIRESSE et Irène MARCOVICI : Density classification on infinite lattices and trees. *Electron. J. Probab.*, 18:no. 51, 22, 2013.

[5] David W. CARMAN, Peter S. KRUUS et Brian J. MATT : Constraints and approaches for distributed sensor network security (final). *DARPA Project report,(Cryptographic Technologies Group, Trusted Information System, NAI Labs)*, 1(1), 2000.

[6] Nazim FATÈS : A guided tour of asynchronous cellular automata. *Journal of Cellular Automata*, 9(5-6):387–416, 2014.

[7] Robert FISCH : Cyclic cellular automata and related processes. *Physica D : Nonlinear Phenomena*, 45(1):19 – 25, 1990.

[8] Robert FISCH, Janko GRAVNER et David GRIFFEATH : *Cyclic Cellular Automata in Two Dimensions*, pages 171–185. Birkhäuser Boston, Boston, MA, 1991.

[9] Nicolas GAUVILLE : Système robuste de diagnostic décentralisé à l'aide d'automates cellulaires simples. Rapport technique, septembre 2018.

[10] Leonardo I. REYES : Greenberg-Hastings dynamics on a small-world network : the collective extinct-active transition. *arXiv preprint arXiv :1505.00182*, 2015.

[11] Antoine SPICHER, Nazim FATÈS et Olivier SIMONIN : Translating discrete multi-agents systems into cellular automata : Application to diffusion-limited aggregation. *In* Joaquim FILIPE, Ana FRED et Bernadette SHARP, éditeurs : *Proc. of ICAART 2009 : Agents and Artificial Intelligence*, pages 270–282. Springer Berlin Heidelberg, 2010.

[12] Antonina TRETYAKOVA, Franciszek SEREDYNSKI et Pascal BOUVRY : Graph cellular automata approach to the maximum lifetime coverage problem in wireless sensor networks. *SIMULATION*, 92(2):153–164, 2016.