

Cycles, double-cycles et modes de mise à jour

Tarek Melliti¹, Mathilde Noual², Damien Regnault¹ et Sylvain Sené^{2,3}

¹Université d'Évry – Val d'Essonne, IBISC, Évry, France

²Université d'Aix-Marseille, CNRS, LIF, Marseille, France

³Institut rhône-alpin des systèmes complexes, IXXI, Lyon, France

Résumé

Qu'il s'agisse de notre vie quotidienne à travers les réseaux sociaux, de ce qui constitue notre essence avec les réseaux de régulation biologique, ou même du monde physique auquel nous appartenons et des systèmes de particules qui le constituent, nous sommes tous concernés de très près par les réseaux d'interactions. Or, les connaissances acquises à leur sujet ne sont que très partielles et il semble aujourd'hui nécessaire, ou tout du moins utile, de mieux appréhender et comprendre leurs propriétés intrinsèques, et notamment les relations entre leur architecture et leur dynamique ainsi que leur rapport au « temps ». En considérant comme modèle général de réseaux d'interactions les réseaux d'automates booléens, nous présentons dans cet article une synthèse de résultats récents sur le rôle de la présence de cycles de rétroaction et de leurs intersections sur leur comportement asymptotique, en mettant notamment en avant l'influence des modes de mise à jour des automates.

Mots-clés : Réseaux d'interactions, réseaux d'automates, systèmes dynamiques discrets, cycles, parallélisme et asynchronisme

1 Introduction

Les réseaux d'interactions (RI) occupent une place importante dans notre vie quotidienne. On le voit aujourd'hui notamment avec l'utilisation massive des réseaux sociaux, dont nous ne faisons qu'entrapercevoir les implications et dont nous ne comprenons toujours pas les mécanismes fondamentaux qui les régissent. Bien sûr, les réseaux sociaux ne sont qu'un exemple parmi tant d'autres de RI. En réalité, à tous les niveaux de nos vies se trouvent de tels réseaux, constitués d'entités qui interagissent localement les unes avec les autres au cours du temps. Ces réseaux admettent, comme conséquence de ces interactions locales, des comportements globaux généralement difficiles à expliquer à partir de la seule connaissance des processus locaux. Parmi les exemples les plus parlants se trouve sans nul doute ce qui est à l'origine de

ce que nous sommes et plus généralement à l'origine de tous les organismes vivants, à savoir les gènes et leur régulation. Il est actuellement admis, en biologie et en médecine, qu'une meilleure compréhension de la régulation génétique est une condition nécessaire à l'amélioration de notre connaissance du vivant, qui pourrait nous permettre d'accéder à des thérapies toujours plus efficaces car plus précises et ciblées.

Bien que les aspects applicatifs liés aux RI aient évidemment une importance réelle et « palpable », et progressent quantitativement de façon effrénée depuis une vingtaine d'années, les aspects plus fondamentaux visant à comprendre et analyser les propriétés intrinsèques de ces réseaux s'avèrent à ce jour moins étudiés par la communauté scientifique. Il en ressort que les applications actuelles, et notamment celles émergeant en biologie (qui constitue ici notre domaine applicatif privilégié), sont en attente d'avancées théoriques significatives pour poursuivre leur développement qualitatif. Or, la science informatique, à laquelle s'associent bien entendu les mathématiques discrètes et la bio-informatique considérée ici selon la définition originelle donnée dans [23, 24], à savoir comme « l'étude des processus informatiques dans les systèmes biotiques », semble particulièrement adaptée pour répondre aux besoins émanant des applications. En effet, les réseaux réels, de par les entités et les interactions qui les composent, sont naturellement discrétisables et peuvent donc assez aisément être représentés par des modèles informatiques qui en capturent l'essence grâce à un niveau d'abstraction élevé. Là où la modélisation mathématique traditionnelle (continue) s'attache principalement aux caractéristiques quantitatives des réseaux, l'intérêt de la modélisation informatique discrète provient de la nature qualitative des questions qu'elle soulève, qui permet de rendre compte que les éléments centraux des réseaux ne sont pas les entités elles-mêmes mais plutôt les interactions qui les lient.

Cet article propose justement d'adopter ce point de vue de l'informatique et plus précisément de l'informatique fondamentale pour étudier certaines des propriétés des RI. Ce choix possède l'avantage de revenir aux origines de l'informatique moderne, avec notamment les travaux des années 1940 sur les automates cellulaires [61] et les réseaux de neurones formels [35] qui se sont fortement inspirés des processus naturels et qui ont permis d'établir les premiers liens entre l'informatique et la biologie. Par ailleurs, les travaux de McCulloch et Pitts sont les premiers à avoir introduit les réseaux d'automates comme modèle de RI, en l'occurrence neuronales. Ce modèle est justement celui choisi dans ce papier comme modèle général des RI (nous en discuterons les raisons plus tard).

Au delà des réseaux d'automates, en particulier booléens qui seront étudiés du point de vue fondamental, cet article propose de mettre en exergue le rôle primordial des cycles de rétro-action, présents dans leur architecture, sur leurs propriétés dynamiques asymptotiques. Au travers de méthodes issues de la théorie des systèmes dynamiques discrets, de la combinatoire énuméra-

tive, de l’algorithmique et de la complexité, ce travail s’attachera à la richesse comportementale des cycles et de leurs intersections (nous n’évoquerons ici que leurs intersections tangentielles). Un autre axe abordé dans ce travail, connu comme étant l’un des plus importants à ce jour des points de vue fondamental et appliqué, sera l’influence des modes de mise à jour sur ces objets particuliers, qui permettent de définir la manière dont les états des automates d’un réseau sont mis à jour en fonction du temps discret. Étant donné un réseau, il existe une infinité de modes de mise à jour pouvant lui être associé. Ici, nous nous attacherons essentiellement aux deux modes les plus utilisés dans la littérature, le mode parallèle et le mode asynchrone. Nous montrerons que leurs influences respectives sur les réseaux induisent des différences profondes sur la dynamique des systèmes étudiés. Enfin, soulignons que cet article est une synthèse de résultats obtenus récemment sur les RI. En conséquence, les résultats ne sont pas donnés de manière exhaustive, il ne sont pas non plus détaillés, en particulier leurs démonstrations, ces dernières étant accessibles dans [36, 40, 53].

À la suite d’une présentation de l’histoire des réseaux d’automates dans le contexte de l’informatique fondamentale moderne et des liens que cette dernière a toujours entretenus avec la biologie, les principales définitions et notations seront explicitées. Ensuite, une section présentera succinctement les résultats fondateurs, obtenus dans les années 1980, mettant en évidence l’importance capitale du rôle des cycles de rétroaction sur l’expressivité dynamique a richesse comportementale des RI. Ceci nous amènera à nous focaliser sur ces cycles, sur leurs propriétés dynamiques et leur richesse comportementale lorsqu’ils sont soumis aux modes de mise à jour parallèle et asynchrone. Enfin, avant de conclure, nous porterons notre attention sur les intersections tangentielles de cycles, que nous appelons plus simplement « double-cycles ».

2 Réseaux d’automates : entre informatique et biologie fondamentales

2.1 La biologie, une source d’inspiration pour l’informatique moderne

Généralement, lorsque nous nous intéressons à l’histoire de l’informatique dite moderne, nous remontons aux années 1930 qui ont vu se développer les paradigmes de calcul classiques tels que les fonctions récursives de Herbrand et Gödel dont les principaux travaux sont accessibles dans [22], le lambda-calcul développé initialement dans [6] et également les machines de Turing [59]. Or, ne penser l’informatique moderne qu’aux travers de ces paradigmes consiste à faire l’impasse sur toute une famille de modèles certes moins conventionnels mais sur lesquels reposent bon nombre de développements de l’informatique que nous connaissons aujourd’hui. Il s’agit de la

famille des réseaux d'automates que l'on doit essentiellement à McCulloch et Pitts, et von Neumann, dont les premiers éléments de la théorie datent des années 1940. En se fondant sur les avancées de l'époque, les travaux originaux [35, 61] mettent en exergue une volonté de développer la science du calcul tout en s'inspirant et en faisant progresser la modélisation de phénomènes naturels biologiques. Ainsi, McCulloch et Pitts introduisent le modèle des *réseaux de neurones formels* qui fournit une abstraction des interactions neuronales. Ils montrent notamment que la logique propositionnelle peut représenter des événements neuronaux et que ces réseaux peuvent être considérés, dans une certaine mesure, comme un modèle de calcul universel. Un peu plus tard, vers la fin des années 1940, von Neumann développe les *automates cellulaires* afin de « comparer les automates naturels et artificiels » et d'« abstraire la structure logique de la vie ». Il résulte de ses travaux la construction du premier automate cellulaire auto-reproducteur et universel.

Les réseaux de neurones formels et les automates cellulaires constituent le socle fondateur de la théorie des réseaux d'automates, un *réseau d'automates* se définissant « grossièrement » comme un ensemble d'entités (des automates) qui interagissent les unes avec les autres, selon des fonctions de transition locales aux entités, suivant un temps discret. Les deux différences majeures qui distinguent ces deux modèles portent sur le nombre d'entités en interaction et la nature du réseau sur lequel les entités sont placées. En effet, là où les automates cellulaires se définissent par défaut avec un nombre infini d'entités (ou cellules) placées sur un réseau régulier et homogène (en général \mathbb{Z}^d , avec $d > 0$), les réseaux de neurones formels (aussi appelés réseau d'automates booléens à seuil) se définissent quant à eux avec un nombre fini d'entités placées sur un réseau irrégulier et hétérogène. Ces différences confèrent à chacun de ces modèles des caractéristiques propres qui ont permis des avancées fortes en informatique.

De par leur nature infinie, l'état de l'art montre que les automates cellulaires ont été majoritairement étudiés pour leurs propriétés de calculabilité. Parmi elles, on retrouve l'universalité de l'automate auto-reproducteur de von Neumann [61], du jeu de la vie de Conway [4] et de l'automate cellulaire élémentaire n°110 [7]. Aussi, en 1971, Smith démontra que toute machine de Turing pouvait être simulée par un automate cellulaire défini sur \mathbb{Z} . Par ailleurs, les automates cellulaires se sont avérés être de bons outils mathématiques pour étudier le fonctionnement parallèle des ordinateurs, dont ils furent à l'origine des architectures systoliques [33]. Enfin, dans le cadre de la théorie des systèmes dynamiques, le souhait de comprendre leur richesse comportementale a été à l'origine d'études de complexité [62, 34]. Les réseaux de neurones formels ont également amené leur lot d'avancées. Dans leur papier originel de 1943, McCulloch et Pitts ont démontré qu'ils permettaient de simuler n'importe quelle fonction booléenne. Kleene a ensuite repris ces travaux et, en se fondant sur leur nature finie, a prouvé que les

langages reconnus par ces objets sont rationnels, ce qui lui a par ailleurs permis d'introduire ce concept de rationalité des langages et d'automates finis [32]. Des conditions de caractérisation comportementales ont également été données et des méthodes algébriques ont alors été développées dans ce cadre [25, 15, 20, 10]. Cette dernière référence met notamment en avant des liens forts entre ces réseaux et le modèle booléen de réseaux de régulation génétique introduit dans [28, 27], qui est à l'origine de nombreux travaux mettant en emphase l'intérêt revêtu par l'informatique dans le contexte de recherches en biologie théorique.

2.2 L'informatique, une source méthodologique pour la biologie moderne

La compréhension des mécanismes de la régulation biologique, dans toute leur diversité, est l'un des grands problèmes actuels en biologie moléculaire, qui a notamment été mis en évidence par Jacob et Monod au début des années 1960, en particulier dans [26, 38]. Or, dès la fin des années 1960, un constat est partagé par deux biologistes, Kauffman (biochimiste et biophysicien) et Thomas (biochimiste et généticien, à l'origine de la découverte des molécules d'ADN et d'ARN). Les méthodes de traitement habituelles issues de la biologie moléculaire ne sont pas adaptées pour traiter un tel problème au niveau génétique. Selon eux, la nature expérimentale des méthodes propres à la biologie ne peut apporter à elle seule de réponse à ce problème. Sur la base des travaux de Delbrück selon lesquels il existe des liens entre les types cellulaires différenciés et les attracteurs des modèles théoriques de réseaux [11], Kauffman et Thomas proposent d'utiliser les mathématiques discrètes pour dépasser la simple connaissance observationnelle des systèmes de régulation, en prônant en un sens que la biologie doit se diriger vers des approches plus générales et systématiques du vivant. Il en résulte deux articles qui organiseront et fédéreront dès lors tout un pan de recherche à la frontière des mathématiques discrètes et de la biologie théorique [28, 55].

Kauffman est ainsi le premier à avoir proposé un modèle de réseaux de régulation génétique, sur la base des réseaux de neurones formels. Ce modèle est connu sous le nom des réseaux booléens, et constitue une formalisation de la régulation où les gènes sont les sommets d'un graphe construit aléatoirement et interagissent au cours du temps (discret) au travers de fonctions locales de transition booléennes, considérant ainsi que les gènes peuvent être exprimés ou non, c'est-à-dire transcrits ou non [29]. Originellement, ce modèle est fondé sur deux hypothèses fortes : les interactions reposent sur une architecture de graphes k -réguliers ; l'évolution est parfaitement synchrone. Des relaxations de ces hypothèses ont été réalisées par la suite [1, 16], et ont donné lieu à des applications à des problèmes biologiques comme l'analyse du comportement du réseau de régulation de la levure [30]. En 1973, Thomas s'oppose à l'hypothèse de parallélisme et développe une autre méthode se

voulant plus proche de la « réalité génétique » [55]. L'idée générale qui s'en dégage est une représentation de la causalité des régulations génétiques sous forme de systèmes de transitions d'états asynchrones, une représentation des réseaux par des graphes signés selon la nature promotrice ou inhibitrice des interactions. Dans [56], Thomas introduit deux conjectures fondamentales, démontrées dans le cadre discrets dans [46, 45]. La première (resp. la seconde) énonce que la présence d'un *cycle positif* (resp. d'un *cycle négatif*), composé d'un nombre pair (resp. impair) d'arêtes inhibitrices, dans l'architecture du réseau est nécessaire à la multi-stationnarité dynamique (resp. à l'existence d'un régime limite oscillant). Au delà des recherches théoriques, cette méthode a été largement appliquée en biologie, comme par exemple à la réponse immunitaire [31] ou encore à l'infection par le phage λ d'*Escherichia coli* [58, 54].

Bien qu'ayant été initiées par des scientifiques issus de la biologie, ces deux visions mettent en exergue les réseaux d'automates et portent ainsi la marque de l'informatique et des mathématiques discrètes. Sur ce point justement, Robert (mathématicien) a été sans nul doute précurseur en s'intéressant dès la fin des années 1960 aux réseaux d'automates d'un point de vue plus fondamental et formel. En effet, là où Kauffman et Thomas ont fait des choix « arbitraires » quant aux manières de mettre à jour les automates au cours du temps, Robert s'intéresse aux modes de mise à jour en tant que tels et à leurs influences sur les comportements des réseaux. Il formalise notamment le concept des *itérations blocs-séquentielles* et des *itérations chaotiques* [48, 49, 50, 51], qui permettent en particulier d'obtenir des modes de mise à jour en partie synchrones et asynchrones. Ceci prend tout son sens en biologie théorique puisqu'aucun argument biologique ne permet aujourd'hui de définir l'organisation temporelle des régulations génétiques. De plus, les travaux menés par Robert et ses collaborateurs ont permis de fixer des bases théoriques solides (simples et générales) pour l'étude comportementale des réseaux d'automates [17, 8, 18, 19] tout en gardant en vue leurs capacités représentationnelles pour la biologie [12, 9, 13]. Dans ce cadre théorique, Robert démontre le caractère essentiel des cycles dans les études sur les propriétés comportementales intrinsèques des réseaux par le théorème stipulant que tout réseau acyclique a un comportement trivial et n'admet à l'asymptote temporelle qu'un unique point fixe. Là encore, les travaux de Robert ont trouvé de nombreuses applications en biologie, parmi lesquelles la modélisation du contrôle génétique du développement de la fleur d'*Arabidopsis thaliana* [37] et l'étude de l'invagination ventrale lors de la gastrulation chez la drosophile [3].

2.3 Les réseaux d'automates booléens, à la fois simples et complexes

D'un point de vue général, les réseaux d'automates peuvent être utilisés pour modéliser tout système qui satisfait les trois propriétés suivantes :

- il est composé d'entités distinctes qui interagissent ensemble ;
- chaque entité se caractérise par une quantité variable, qui a précisément vocation à être traduite en termes d'états de l'automate correspondant dans le modèle ;
- les événements subis par le système, comme les mécanismes qui en sont à l'origine, ne sont observables ni directement ni intégralement de manière certaine. Seules leurs conséquences le sont, à savoir des changements entièrement accomplis.

Ces trois propriétés n'imposent que très peu de restrictions sur l'ensemble des systèmes que l'on peut abstraire et ainsi modéliser par des réseaux d'automates. Ces objets théoriques sont par conséquent des modèles génériques d'une très grande variété de systèmes réels. Il est donc assez aisé de comprendre les raisons qui ont poussé les scientifiques à les utiliser et à ne cesser de les étudier dans le contexte de la « bio-informatique fondamentale ».

Revenons sur la quantité variable des entités mentionnée plus haut. Pour être traduite en termes d'états d'automates, elle appelle à une toute première formalisation. Celle-ci consiste à choisir si ce qui nous intéresse dans la variation de cette quantité est de nature booléenne, discrète ou continue. À titre d'illustration, prenons l'exemple de la régulation génétique et choisissons comme quantité variable l'action d'un gène. Si, dans l'action de ce gène, ce qui nous intéresse est son expression (et son inexpression), alors l'état de l'automate choisi pour le modéliser sera naturellement booléen. Si ce sont les différentes façons que ce gène a d'agir sur les autres éléments du système qui nous intéressent, l'on pourra choisir de faire correspondre un état de l'automate pour chacune d'elles. Cela induit un formalisme discret qui peut évidemment être encodé sans perte dans un formalisme booléen, étant donné qu'un automate à k états peut être représenté par $\log_2(k)$ automates booléens. Enfin, si l'on mesure l'action du gène au moyen de la concentration des protéines qu'il produit, le formalisme continu s'avère le plus naturel. Il apporte en revanche un caractère quantitatif. Si cet aspect n'est pas celui désiré, la tendance sera alors d'approximer la fonction de concentration protéique par intervalles afin de retomber dans un cadre discret, voire booléen en ne considérant que les concentrations extrémales par exemple. L'on peut donc accorder des statuts différents au contexte booléen selon qu'on le voit comme une modélisation directe de la réalité ou comme une approximation ou un encodage d'une modélisation intrinsèquement continue ou discrète. Relevons que la modélisation booléenne directe est cohérente avec le choix de se focaliser sur les changements d'états des automates plutôt que sur leurs états eux-mêmes. Par analogie à la mécanique, si l'on voit les automates comme

des moteurs à explosion, l'intérêt porte plus sur le fait qu'un moteur soit capable de passer de l'état « arrêté » à l'état « en marche » (et inversement) que sur la quantité d'électricité fournie par la batterie pour démarrer ou que sur celle libérée par les bougies pour provoquer l'explosion et initier le mouvement. Sous cette hypothèse, l'abstraction booléenne est nécessaire et suffisante. Par ailleurs, afin de nous replacer dans le contexte de la modélisation en biologie, il faut souligner que le discours des biologistes est généralement empreint d'éléments syntaxiques de la logique propositionnelle. Il n'est en effet pas rare d'entendre des phrases telles que : « en l'absence du répresseur α , le gène β s'exprime » ou encore « si les produits des gènes α et β forment un complexe, ce dernier favorise l'expression du gène γ alors que ces gènes tendent à inhiber son expression lorsqu'ils sont sous forme monomérique. » Cette syntaxe s'accorde elle aussi parfaitement avec une modélisation directe de la réalité dans le formalisme booléen.

Par ailleurs, les réseaux d'automates booléens tirent de leur simplicité de définition d'autres bénéfices intéressants. Ils fournissent notamment un cadre de travail aux contours clairement définis, idéal pour aborder des problèmes fondamentaux autour de la modélisation des systèmes d'entités en interaction. Étant donné la variété de leur nature et l'état actuel de nos connaissances, les problèmes en question ne pourraient présentement bénéficier de cadres significativement plus élaborés. Cela mènerait inévitablement à délayer les questions premières et à déstructurer le problème posé en attirant l'attention sur des problèmes annexes induits par le jeu de paramètres à considérer et non intrinsèquement inclus dans le problème initial. Pour ces problèmes, au contraire, les réseaux d'automates booléens offrent le juste nécessaire et facilitent la manipulation d'un concept minimal de causalité, qui s'enracine sur la notion de changement d'état. Leur mérite repose donc dans la fiabilité des informations qu'ils fournissent potentiellement, délivrée par leur très haut niveau d'abstraction qui permet par ailleurs d'obtenir des lois générales qui restent valides dans des contextes plus spécifiques. En d'autres termes, il est crucial de comprendre que cette simplicité de définition ne les rend pas pour autant « simplistes », et n'enlève rien de leur capacité de modéliser des phénomènes complexes qu'ils permettent d'analyser qualitativement avec une finesse étonnante.

3 Définitions et notations générales

Informellement, un réseau d'automates booléens (RAB) met en œuvre des entités abstraites interagissant les uns avec les autres dont les états, soit 0 (inactif) soit 1 (actif), peuvent changer au cours d'un temps discret selon l'influence des états des autres entités du réseau. Cette section vise à présenter le formalisme de ce modèle, en en donnant les principales définitions et notations utiles dans la suite de l'article.

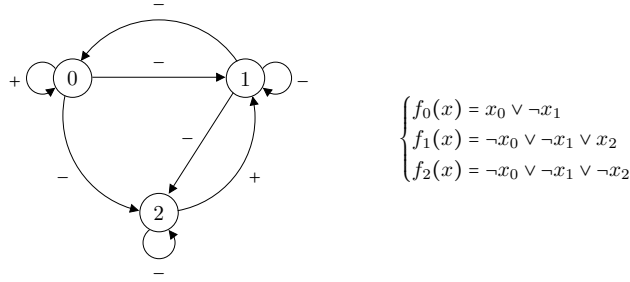


FIGURE 1 – Exemple de RAB de taille 3 : (gauche) son graphe d’interactions, (droite) l’ensemble de ses fonctions locales de transition.

3.1 Réseaux d’automates booléens

Soit $\mathbb{B} = \{0, 1\}$ et $V = \{0, \dots, n-1\}$ un ensemble de n automates booléens tels que $\forall i \in V$, $x_i \in \mathbb{B}$ représente l’état de i . Une configuration x d’un RAB \mathcal{R} de taille n affecte une valeur de \mathbb{B} à chacun des automates de V et est classiquement notée comme un vecteur $x = (x_0, \dots, x_{n-1})$ sommet du n -cube \mathbb{B}^n , ou encore comme un mot binaire. Formellement, un RAB \mathcal{R} dont l’ensemble d’automates est V est un ensemble ordonné de n fonctions booléennes, tel que $\mathcal{R} = (f_i : \mathbb{B}^n \rightarrow \mathbb{B} \mid i \in V)$. Étant donné $i \in V$, f_i est la *fonction locale de transition* de l’automate i . Elle prédétermine son évolution depuis n’importe quelle configuration x : si i est mis à jour dans la configuration x au temps t , il passe de l’état $x_i(t)$ à l’état $f_i(x(t)) = x_i(t+1)$.

Soit $s : \mathbb{B} \rightarrow \mathbb{1}$, avec $\mathbb{1} = \{-1, 1\}$, définie telle que $s(b) = b - \neg b$, la fonction permettant de convertir un nombre booléen en entier signé dans $\mathbb{1}$. Dans cet article, nous portons une attention particulière aux changements d’état des automates, qui nous amène à introduire les notations suivantes $\forall i \in V, \forall x \in \mathbb{B}^n$:

$$\forall i \in V, \bar{x}^{\{i\}} = (x_0, \dots, x_{i-1}, \neg x_i, x_{i+1}, \dots, x_{n-1}).$$

$$\forall W \subseteq V, \bar{x}^{W \cup \{i\}} = \overline{\bar{x}^W}^i.$$

Le *signe d’une interaction* de i à j dans la configuration x est défini par $\text{sign}_x(i, j) = s(x_i) \cdot (f_j(x) - f_j(\bar{x}^{\{i\}}))$. Les interactions *effectives* dans x appartiennent à $A(x) = \{(i, j) \in V \times V \mid \text{sign}_x(i, j) \neq 0\}$. De là, on définit le *graphe d’interactions* de \mathcal{R} comme étant le graphe orienté $G = (V, A)$, où $A = \bigcup_{x \in \mathbb{B}^n} A(x)$ est l’ensemble des interactions. Dans ce papier, les RAB abordés (cf. section 3.3) sont particuliers dans le sens où leurs graphes d’interactions sont simples, à savoir qu’il ne peut y avoir qu’une unique interaction signée $(i, j) \in A$. Si elle est signée positivement (resp. négativement), on dit qu’elle est activatrice (resp. inhibitrice) et l’état de j tend à mimer (resp. à s’opposer à) celui de i . Dans la suite, les graphes d’interactions seront signés par commodité de lecture (cf. figure 1).

3.2 Modes de mise à jour et graphe de transitions

Afin de déterminer les comportements possibles d'un RAB, il est indispensable de spécifier la manière donc les états des automates (ou par abus de langage les automates) sont mis à jour dans le temps. C'est cette spécification que nous appelons *mode de mise à jour*. Le point de vue le plus général consiste à considérer toutes les possibilités. Cela revient à voir l'évolution des réseaux comme un système dynamique discret associé à une relation de manière que, pour chaque configuration, $2^n - 1$ transitions sortant soient prises en compte, à savoir une transition pour chaque sous-ensemble d'automates dont les états peuvent être mis à jour. Plus précisément, $\forall W \neq \emptyset \subseteq V$, on définit la fonction de mise à jour $F_W : \mathbb{B}^n \rightarrow \mathbb{B}^n$ tel que :

$$\forall x \in \mathbb{B}^n, \forall i \in V, F_W(x)_i = \begin{cases} f_i(x) & \text{si } i \in W, \\ x_i & \text{sinon.} \end{cases}$$

Ainsi, pour le mode de mise à jour le plus général, nommé *mode de mise à jour élémentaire*, le comportement global du réseau est donné par le *graphe de transitions élémentaire* $\mathcal{G}_e = (\mathbb{B}^n, T_e)$, où $T_e = \{(x, F_W(x)) \mid x \in \mathbb{B}^n, W \neq \emptyset \subseteq V\}$, introduit dans [40, 53].

Les transitions $(x, F_{\{i\}}(x))$ qui implique la mise à jour d'un unique automate $i \in V$ sont appelées *transitions asynchrones*. Les transitions $(x, F_W(x))$, avec $|W| > 1$, qui induisent la mise à jour de plusieurs automates sont quant à elles appelées *transitions synchrones*. Le sous-graphe $\mathcal{G}_a = (\mathbb{B}^n, T_a)$ de \mathcal{G}_e dont l'ensemble des arcs $T_a = \{(x, F_{\{i\}}(x)) \mid x \in \mathbb{B}^n, i \in V\}$ égale l'ensemble des transitions asynchrones du réseau est appelé le *graphe de transitions asynchrone*. Ce graphe définit le *mode de mise à jour asynchrone* selon lequel, dans chaque configuration, seulement n transitions sont considérées, une pour chaque automate. Ce mode de mise à jour a été largement utilisé dans les études de Thomas et de ses collaborateurs [42, 43, 47, 46, 45, 56, 57]. À titre d'illustration, le graphe de transitions asynchrone du réseau de la figure 1 est présenté dans la figure 2. Parce qu'il est le plus « naturel » mathématiquement parlant quand seules les fonctions locales de transition sont connues, et parce qu'il permet de donner des intuitions pertinentes au travers de graphes de transitions de plus petites tailles, le *mode de mise à jour parallèle* occupe une place très importante dans la littérature sur les systèmes dynamiques discrets en général. Lorsqu'un réseau évolue en parallèle, tous ses automates mettent à jour leur état à chaque étape de temps. Autrement dit, le *graphe de transitions parallèle* d'un réseau est $\mathcal{G}_p = (\mathbb{B}^n, T_p)$, où $T_p = \{(x, F_V(x)) \mid x \in \mathbb{B}^n\}$ est le sous-ensemble des transitions parfaitement synchrones de T_e . Le graphe de transitions parallèle du réseau de la figure 1 est illustré dans la figure 2.

Avant de donner les définitions et notations propres aux cycles d'interactions et à leur tangence, nous précisons celles propres aux comportements

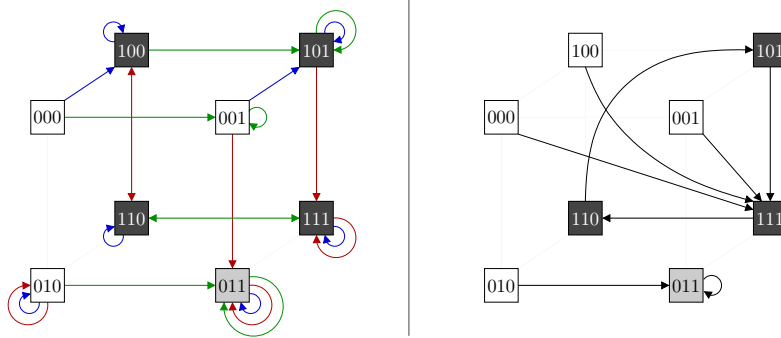


FIGURE 2 – Graphes de transitions (gauche) asynchrone et (droite) parallèle du RAB défini dans la figure 1.

dynamiques des réseaux et qui vont être essentielles dans la suite du papier. Pour ce faire, considérons un RAB arbitraire \mathcal{R} de taille n dont le graphe de transitions est noté $\mathcal{G} = (\mathbb{B}^n, T)$ et $x \in \mathbb{B}^n$ l'une de ses configurations possibles. Une *trajectoire* de x est n'importe quel chemin dans \mathcal{G} qui part de x . Une composante fortement connexe de \mathcal{G} qui n'admet aucune transition sortante représente un comportement asymptotique de \mathcal{R} , que l'on désigne classiquement comme étant un *attracteur* de \mathcal{R} . Une configuration de \mathbb{B}^n qui appartient à un attracteur est une *configuration récurrente*. Étant donné un attracteur, sa *longueur* est le nombre des configurations récurrentes qui le composent. Un attracteur de longueur 1 (resp. de longueur strictement supérieure à 1) est une *configuration stable* (resp. une *oscillation stable*) de \mathcal{R} . Lorsqu'un RAB est soumis à un mode de mise à jour déterministe, comme le mode parallèle, les attracteurs sont des cycles simples et l'on préfère le terme *période* pour évoquer leur longueur. Enfin, on définit le *temps de convergence d'une configuration x* comme la longueur de sa trajectoire la plus courte qui lui fait atteindre une configuration récurrente et le *temps de convergence du réseau* comme le plus grand des temps de convergence de toutes ses configurations de \mathbb{B}^n . Dans les graphes de transitions présentés dans ce papier, par convention, nous associons la couleur gris clair aux configurations stables et la couleur gris foncé aux configurations récurrentes appartenant à des oscillations stables. Ainsi, dans la figure 2, on observe que la configuration 011 est stable pour les modes de mise à jour asynchrone et parallèle. Ceci est une conséquence directe du fait qu'une configuration stable est conservée quel que soit le mode de mise à jour étant donné qu'elle correspond au vecteur des points fixes locaux des fonctions locales de transition. En revanche, on observe que le graphe de transitions asynchrone admet une oscillation stable de taille 4 alors que le mode de mis à jour parallèle crée une oscillation stable de taille 3. Ceci illustre la non-conservation des oscillations stables en général d'un mode de mise à jour à l'autre, qui constitue un champ d'études actuel de premier plan dans le contexte des RI et des RAB.

3.3 Cycles et double-cycles

Comme évoqué dans l'introduction, nous nous focalisons dans cet article sur les cycles et les double-cycles (abrégés respectivement par CAB – cycles d'automates booléens – et DCAB – double-cycles d'automates booléens). Les résultats fondateurs qui nous poussent à développer toujours plus les recherches sur ces motifs d'interactions sont présentés dans la section suivante.

Un CAB \mathcal{C}_n est un RAB de taille n dont le graphe d'interactions $G = (V, A)$ est un cycle, au sens de la théorie des graphes. V est naturellement assimilé à $\mathbb{Z}/n\mathbb{Z}$, de manière qu'en considérant deux automates i et j de V , $i + j$ représente $i + j \bmod n$. Ainsi, un CAB \mathcal{C}_n se définit comme un ensemble ordonné de fonctions locales de transition d'arité 1 qui sont telles que : $\forall i \in V$, $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$, et soit $f_i(x) = x_{i-1}$ soit $f_i(x) = \neg x_{i-1}$. Soulignons qu'il existe deux types de CAB, les positifs et les négatifs. Un CAB est un *cycle positif* \mathcal{C}^+ (resp. un *cycle négatif* \mathcal{C}^-) s'il est composé d'un nombre pair (resp. d'un nombre impair) d'interactions inhibitrices.

Un DCAB \mathcal{D}_n , avec $n = \ell + r - 1$, est un RAB de taille n composé de deux CAB \mathcal{C}_ℓ et \mathcal{C}_r qui s'intersectent tangentiellement en un automate, l'automate 0. Dans la suite, pour des raisons de clarté, nous préférons la notation $\mathcal{D}_{\ell,r}$. L'ensemble des automates du cycle \mathcal{C}_ℓ est $V^\mathcal{L} = \mathbb{Z}/\ell\mathbb{Z} = \{0, \dots, \ell - 1\}$ et celui du cycle \mathcal{C}_r est $V^\mathcal{R} = \{0\} \cup \{\ell - 1 + i \mid i \neq 0 \in \mathbb{Z}/r\mathbb{Z}\}$. Dans un DCAB, par définition, toutes les fonctions locales de transition sont d'arité 1 excepté celle de l'automate 0 qui est d'arité 2. Dans ce travail, nous considérons uniquement des DCAB localement monotone, ce qui induit que la fonction f_0 est définie comme $f_0(x) = f_0^\mathcal{L}(x) \diamond f_0^\mathcal{R}(x) = x_{\ell-1} \diamond x_{n-1}$, où $\diamond \in \{\wedge, \vee\}$. Notons que le choix de l'opérateur \diamond ne change que la position des configurations sur les trajectoires. Autrement dit, quel que soit le mode de mise à jour choisi, étant donnés deux DCAB $\mathcal{D}_{\ell,r}$ et $\mathcal{D}'_{\ell,r}$ de même taille tels que $f_0(x) = f_{\ell-1}(x) \wedge f_{n-1}(x)$ et $f'_0(x) = f'_{\ell-1}(x) \vee f'_{n-1}(x)$, leurs graphes de transitions respectifs seront identiques à un isomorphisme des configurations près. De plus, il est trivial de déterminer l'un à partir de l'autre, en remplaçant les configurations par leurs opposées. Par conséquent, dans la suite de ce papier, pour insister sur cette propriété, nous utiliserons sans perte de généralité $f_0(x) = f_{\ell-1}(x) \vee f_{n-1}(x)$ (resp. $f_0(x) = f_{\ell-1}(x) \wedge f_{n-1}(x)$) pour les DCAB soumis au mode de mise à jour parallèle (resp. asynchrone). Puisqu'ils sont composés de deux CAB, il est aisé de voir qu'il existe trois types de DCAB distincts. Un DCAB *positif* est composé de deux CAB positifs et est noté $\mathcal{D}_{\ell,r}^{+,+}$; un DCAB *négatif* est composé de deux CAB négatifs et est noté $\mathcal{D}_{\ell,r}^{-,-}$; DCAB *mixte* est composé d'un CAB négatif \mathcal{C}_ℓ^- et d'un CAB positif \mathcal{C}_r^+ , et est noté $\mathcal{D}_{\ell,r}^{-,+}$.

Enfin, dans [40, 53], les auteurs ont montré que les CAB admettent des représentants canoniques. Un CAB positif (resp. négatif) est dit canonique si son graphe d'interactions ne contient que des interactions activatrices (resp. si son graphe d'interactions admet une unique interaction inhibitrice, repré-

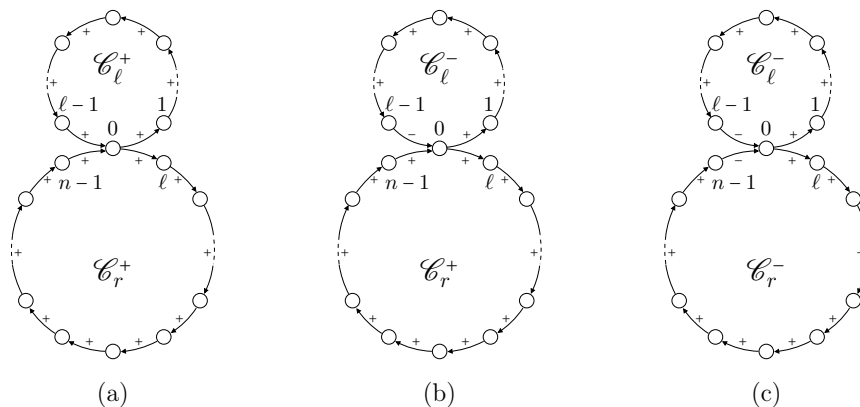


FIGURE 3 – Exemples de double-cycles canoniques : (a) un DCAB positif, (b) un DCAB mixte et (c) un DCAB négatif.

sentée par l’arc $(n - 1, 0) \in A$. Les DCAB canoniques sont quant à eux les compositions de CAB canoniques, représentés dans la figure 3.

4 Résultats fondateurs sur les cycles

Les travaux de Robert [50, 51] et de Thomas [56] ont mis en exergue trois résultats fondamentaux qui explicitent le rôle primordial que jouent les cycles sur la richesse comportementale des RI.

Théorème 1. [50, 51] *Soit \mathcal{R} un réseau d’automates et G son graphe d’interactions. Si G est acyclique, alors toutes les trajectoires de \mathcal{R} convergent vers une même et unique configuration stable.*

Ce théorème est particulièrement intéressant pour deux raisons. Tout d’abord, il souligne que les cycles sont des conditions nécessaires aux RI pour qu’ils admettent une dynamique complexe. Ensuite, il ne pose pas de contrainte particulière, que ce soit sur la nature des automates (booléens ou multi-valués) ou sur le mode de mise à jour. Quant à l’idée générale de la démonstration, elle repose sur un raisonnement pas récurrence sur les profondeurs des automates du graphe et sur le fait que, puisque G est acyclique, il admet des automates sources gouvernés par des fonctions locales de transition constantes.

Théorème 2. [43, 46, 56] *Soit \mathcal{R} un réseau d’automates et G son graphe d’interactions. Sous le mode de mise à jour asynchrone, la présence d’un cycle positif est nécessaire pour que la dynamique de \mathcal{R} admette plusieurs configurations stables.*

Ce deuxième théorème éclaire le rôle des cycles positifs sur la capacité des RI à se stabiliser de plusieurs façons. Bien qu’il ait originellement été énoncé

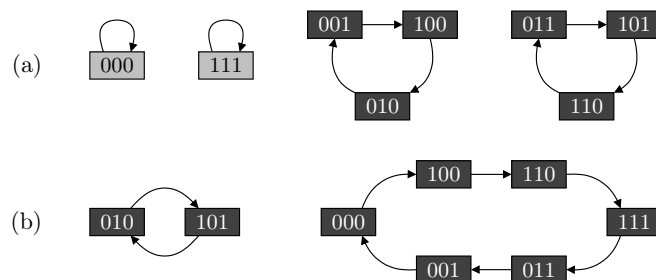


FIGURE 4 – Graphes de transitions parallèles : (a) du CAB canonique positif \mathcal{C}_3^+ et (b) du CAB canonique négatif \mathcal{C}_3^- .

et démontré sous l’hypothèse du mode asynchrone, ce résultat est également général à tout mode de mise à jour [40, 53]. La preuve générale part du résultat de [46] et se fonde sur un raisonnement par l’absurde sous l’hypothèse de l’absence de cycle positif. Dans ce cas, soit le graphe d’interactions est acyclique et le théorème 1 s’applique, soit il admet au moins un cycle négatif et l’on démontre que de tels cycles, quel que soit le mode de mise à jour, ne peuvent supprimer toutes les instabilités locales au niveau des automates.

Théorème 3. [45, 56] *Soit \mathcal{R} un réseau d’automates et G son graphe d’interactions. Sous le mode de mise à jour asynchrone, la présence d’un cycle négatif est nécessaire pour que la dynamique de \mathcal{R} admette une oscillation stable.*

Quant à ce troisième théorème, il s’avère ne pas être général à tous les modes de mise à jour. Pour s’en convaincre, il suffit de calculer le graphe de transitions parallèle d’un cycle positif arbitraire de taille supérieure à 2 et de remarquer qu’il admet au moins une oscillation stable. Malgré cette absence de généralité, remarquons que ce théorème souligne le rôle singulier des cycles négatifs en lien avec les oscillations dynamiques asymptotiques, comme cela sera précisé plus bas.

Pris ensemble, ces trois théorèmes, dont on peut dire que deux d’entre eux sont des lois dans la mesure où leur généralité en font des sortes de méta-théorèmes de la théorie des RI, mettent en avant que les cycles de rétroaction sont les causes de la complexité dynamique des réseaux. Autrement dit, ils constituent effectivement les sources de leur richesse comportementale et par conséquent de leur expressivité calculatoire, ce qui nous amène naturellement aux sections suivantes, qui exposent les résultats récents majeurs concernant les propriétés dynamiques et combinatoires des CAB et DCAB et les illustrent par des exemples.

5 Dynamique des cycles isolés

5.1 Cycles parallèles

Les premiers éléments sur la dynamique des CAB soumis au mode mise à jour parallèle ont été introduits dans [42]. C'est par la suite dans [14] qu'elle a été entièrement caractérisée. Cette caractérisation a pu être obtenue grâce à une approche mêlant la théorie des systèmes dynamiques discrets et la combinatoire énumérative, particulièrement appropriée à la nature des objets mathématiques en question.

5.1.1 Définitions et notations des quantités pertinentes

Pour décrire les résultats obtenus, nous donnons ci-dessous les définitions et notations utiles. Tout d'abord, étant donné un attracteur de période (minimale) p , on dit que tous les multiples de p sont aussi des périodes de cet attracteur. Ainsi, si $x \in \mathbb{B}^n$ est une configuration récurrente d'un attracteur de période p , alors p est la période de x et de toute autre configuration y telle que $\exists t \in \mathbb{N}, y = F_V^t(x)$. On note $\mathcal{X}(p) = \{x \in \mathbb{B}^n \mid x = F_V^p(x)\}$ l'ensemble des configurations de période p et $X(p) = |\mathcal{X}(p)|$ leur nombre. On définit le plus petit entier ω qui est une période commune à toutes les configurations récurrentes comme l'*ordre* du CAB, dont on dit qu'*il est atteint* lorsqu'il existe un attracteur de période minimale ω .

Considérons la fonction $\mathbf{un} : n \in \mathbb{N} \mapsto 1$ ainsi que la *convolution de Dirichlet*, notée \star [2]. \star est l'opérateur binaire qui, étant données deux fonctions f et g , se définit tel que $f \star g : n \in \mathbb{N}^* \mapsto \sum_{p|n} f(p) \cdot g(n/p)$. L'ensemble des fonctions arithmétiques avec l'addition point à point et la convolution de Dirichlet est un anneau commutatif. L'identité par la multiplication de cet anneau est la fonction $\delta : \mathbb{N}^* \rightarrow \mathbb{N}^*$, définie par $\delta(1) = 1$ and $\forall n > 1, \delta(n) = 0$. L'inverse de la fonction \mathbf{un} pour la convolution de Dirichlet est la fonction de Möbius μ , définie par :

$$\mu : n \in \mathbb{N}^* \mapsto \begin{cases} 0 & \text{si } n \text{ est sans facteur carré parfait} \\ 1 & \text{si } n > 0 \text{ a un nombre pair de facteurs premiers} \\ -1 & \text{si } n > 0 \text{ a un nombre impair de facteurs premiers} \end{cases} .$$

Si $n = \prod_{i=0}^k p_i$, où les p_i sont des nombres premiers distincts, alors $\mu(n) = (-1)^k$. Dans notre contexte, cette fonction présente de l'intérêt au travers de la *formule d'inversion de Möbius* qui est obtenue à partir de $\mathbf{un} \star \mu = \delta$, qui est satisfaite par toutes les fonctions f et g , et qui est telle que : $g = f \star \mathbf{un} \implies f = g \star \mu$. En d'autres termes, on a :

$$\forall n \in \mathbb{N}^*, g(n) = \sum_{p|n} f(p) \implies f(n) = \sum_{p|n} g(p) \cdot \mu(n/p).$$

Une autre fonction particulièrement utile dans la suite est l'*indicatrice d'Euler*, notée ϕ . Étant donné un entier $n \in \mathbb{N}^*$, elle lui associe le nombre d'entiers strictement positifs inférieurs ou égaux à n et premiers avec n , tel que : $\phi(n) = |\{m \in \mathbb{N}^* \mid m \leq n \text{ et } m \text{ est premier avec } n\}|$. À noter qu'il existe une relation entre la fonction de Möbius et l'indicatrice d'Euler. En effet, comme ϕ satisfait $\forall n \in \mathbb{N}^*, n = \phi \star \text{un}(n)$, elle respecte $\phi = \mu \star \text{id}$, où $\text{id} : n \in \mathbb{N}^* \mapsto n$.

En termes de combinatoire, le comportement asymptotique d'un RI peut être décrit par quatre quantités [41] liées les unes aux autres et données ci-dessous. Considérons que p est un diviseur de l'ordre ω du CAB étudié et la fonction $\text{inv} : n \in \mathbb{N}^* \rightarrow 1/n$. On a alors les quantités suivantes :

- le nombre $\mathbf{X}(p)$ de configurations de période p est $\mathbf{X} = \tilde{\mathbf{X}} \star \mathbb{1}$;
- le nombre $\tilde{\mathbf{X}}(p)$ de configurations de période minimale p est $\tilde{\mathbf{X}} = \mathbf{X} \star \mu$;
- le nombre $\mathbf{A}(p) = \tilde{\mathbf{X}}(p)/p$ d'attracteurs de période p est $\mathbf{A} = \text{inv}(\mathbf{X} \star \mu)$;
- Le nombre total $\mathbf{T}(\omega)$ d'attracteurs est $\mathbf{T} = \mathbf{A} \star \mathbb{1} = \text{inv}(\mathbf{X} \star \phi)$.

Les deux dernières quantités correspondent à des formules bien connues dans le cadre des mots de Lyndon et des colliers [5, 21, 52], un *collier* w de longueur n étant un mot binaire circulaire tel que $\forall i \in \mathbb{Z}, w_i = w_{i \bmod n}$. Notons également que la dernière est satisfaite car inv se distribue sur \star . Enfin, remarquons qu'il suffit de calculer \mathbf{X} pour qu'en découlent les valeurs de $\tilde{\mathbf{X}}$, \mathbf{A} et \mathbf{T} .

5.1.2 Résultats

La caractérisation qualitative de la dynamique des CAB peut se résumer par le théorème qui suit et qui présente sous la forme d'un tableau l'ensemble de toutes les quantités présentées plus haut. Pour des raisons de place, les détails et les démonstrations relatifs à ces résultats ne sont pas présentés ici. Le lecteur pourra néanmoins les trouver tous dans [14, 40, 53].

Théorème 4. *L'ordre ω , les nombres $\mathbf{X}(p)$ et $\tilde{\mathbf{X}}(p)$ des configurations de période (minimale) p , où $\mathbf{X}(\omega)$ représente le nombre total de configurations récurrentes, ainsi que le nombre $\mathbf{A}(p)$ d'attracteurs de période p et le nombre \mathbf{T} total d'attracteurs des cycles positifs et négatifs sont les suivants, où $-(p|m)$ vaut 0 si p divise m et 1 sinon :*

	Cycles positifs \mathcal{E}_n^+	Cycles négatifs \mathcal{E}_n^-
Ordre ω	n	$2n$
Nombre de configurations de période p , $p \omega$	$\mathbf{X}^+(p) = 2^p$	$\mathbf{X}_n^-(p) = -(p n) \cdot 2^{\frac{p}{2}}$
Nombre de configurations de période minimale p , $p \omega$	$\tilde{\mathbf{X}}^+(p) = \sum_{d p} \mu\left(\frac{p}{d}\right) \cdot 2^d$ $= \text{OEIS A27375}(p)$	$\tilde{\mathbf{X}}_n^-(p) = \sum_{k p \text{ impair}} \mu(k) \cdot 2^{\frac{p}{2k}}$

	Cycles positifs \mathcal{C}_n^+	Cycles négatifs \mathcal{C}_n^-
Nombre d'attracteurs de période p , $p \omega$	$A^+(p) = \frac{\tilde{\chi}^+(p)}{p}$ $= \text{OEIS A1037}(p)$	$A_n^-(p) = \frac{\tilde{\chi}_n^-(p)}{p}$ $= \text{OEIS A48}(\frac{p}{2})$
Nombre total d'attracteurs	$T^+(\omega) = \frac{1}{n} \sum_{d n} \phi(\frac{n}{d}) \cdot 2^d$ $= \text{OEIS A31}(n)$	$T^-(\omega) = \frac{1}{2n} \sum_{k 2n \text{ impair}} \phi(k) \cdot 2^{\frac{2k}{k}}$ $= \text{OEIS A16}(n)$

Parmi les propriétés particulièrement intéressantes d'un CAB parallèle qui ressortent de ce théorème, il faut remarquer que toutes ses configurations sont récurrentes, comme l'illustre la figure 4 et que le nombre total d'attracteurs, pour une période donnée ou non, est exponentiel au regard de sa taille n . Aussi, en corollaire, pour les deux types de cycles, l'ordre est atteint, et les cycles positifs admettent deux configurations stables x et \bar{x}^V (où x est la configuration dans laquelle tous les automates sont à l'état 0 dans les cycles canoniques). Enfin, un autre point important est que cette étude combinatoire induit la caractérisation complète de la structure des graphes de transitions parallèles des cycles positifs et négatifs.

5.2 Cycles asynchrones

En termes combinatoires, la dynamique des cycles asynchrones est nettement plus simple que celle des cycles parallèles. C'est dans [42] qu'on en trouve la première caractérisation issue des propriétés dynamiques mises en exergue pour les cycles parallèles. L'idée générale de ses résultats se fonde sur les concepts d'instabilité, en considérant qu'un automate d'une configuration est instable lorsque son état change par application de sa fonction locale de transition et qu'une configuration est instable quand elle admet au moins un automate instable. Notamment, les auteurs ont montré que l'asynchronisme permet de réduire temporellement le nombre d'instabilités, jusqu'à atteindre 0 (resp. 1) dans le cas des cycles positifs (resp. négatifs), propriété qui souligne l'existence d'au moins un point fixe (resp. l'absence de point fixe et donc l'existence d'au moins une oscillation stable). Il ressort presque directement de ce travail le théorème 5, illustré par la figure 5.

Théorème 5. *Un cycle positif \mathcal{C}_n^+ admet deux uniques attracteurs qui sont deux configurations stables x et \bar{x}^V (avec $x = (0, \dots, 0)$ dans le cas d'un cycle canonique). Un cycle négatif \mathcal{C}_n^- admet un unique attracteur de longueur $2n$.*

Il est aisé de voir que les résultats énoncés dans ce théorème sont fortement liés à ceux du théorème 4. En effet, outre la propriété élémentaire que les deux configurations stables des cycles positifs asynchrones sont identiques à celles de ces mêmes cycles en parallèle, une propriété corollaire pertinente concerne l'unique oscillation stable de longueur $2n$ d'un cycle négatif \mathcal{C}_n^- ,

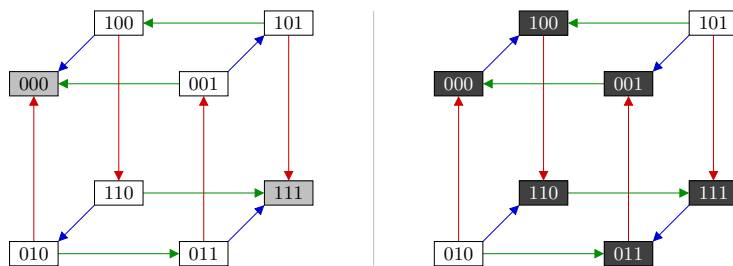


FIGURE 5 – Graphes de transitions asynchrones (les transitions inefficaces sont omises) : (à gauche) du CAB canonique positif \mathcal{E}_3^+ et (à droite) du CAB canonique négatif \mathcal{E}_3^- .

identique à celle de période ω des cycles négatifs parallèles incluant les configurations x et \bar{x}^V qui sont stables pour les cycles positifs.

Enfin, remarquons que cette approche fondée sur les instabilités a été généralisée dans [40, 53] pour démontrer la validité du théorème 2 quel que soit le mode de mise à jour (cf. section 4).

6 Quid des double-cycles ?

Maintenant que les propriétés structurelles et combinatoires des graphes de transitions parallèles et asynchrones des CAB sont établies, nous présentons celles des DCAB, en proposant notamment une approche algorithmique tout à fait nouvelle pour le cas asynchrone.

6.1 Double-cycles parallèles

6.1.1 Définitions et notations

Outre les quantités $\mathbf{X}(p)$, $\tilde{\mathbf{X}}(p)$, \mathbf{A} et \mathbf{T} , dont nous allons nous resservir, nous introduisons ici d'autres définitions et notations, centrales pour caractériser la dynamique des DCAB qui, rappelons-le, sont des CAB interconnectés tangentiellement. En ce sens, étant donné un DCAB quelconque $\mathcal{D}_{\ell,r}^s$, où $s \in \{(+,+), (-,+), (-,-)\}$, deux quantités définies au moyen de ℓ et r seront particulièrement utiles, $\Delta = \text{PGCD}(\ell, r)$ et $\Delta_p = \text{PGCD}(\Delta, p)$.

Aussi, puisque les résultats font appel dans ce contexte à la théorie des mots, nous introduisons les suites de Lucas et de Perrin, qui permettent de compter le nombre de configurations récurrentes. La *suite de Lucas* $(L(n))_{n \in \mathbb{N}^*}$ (OEIS A204) est définie par $L(1) = 1$, $L(2) = 3$ et $L(n) = L(n-1) + L(n-2)$, $\forall n > 2$, et compte le nombre de colliers de taille n sans le facteur 00. La *suite de Perrin* $(P(n))_{n \in \mathbb{N}}$ (OEIS A1608) est définie par $P(0) = 3$, $P(1) = 0$, $P(2) = 2$ et $P(n) = P(n-2) + P(n-3)$, $\forall n > 2$, et compte le nombre de colliers de taille n sans les facteurs 00 et 111.

6.1.2 Résultats

Sur la base des définitions et notations précédentes, le théorème 6 suivant donne la caractérisation qualitative de la dynamique des différents types de DCAB, dont les détails des démonstrations sont présentés dans [39, 40, 53].

Théorème 6. *L'ordre ω , les nombres $\mathbf{X}(p)$ et $\tilde{\mathbf{X}}(p)$ des configurations de période (minimale) p , où $\mathbf{X}(\omega)$ représente le nombre total de configurations récurrentes, ainsi que le nombre $\mathbf{A}(p)$ de comportements asymptotiques de période p et le nombre \mathbf{T} total de comportements asymptotiques des double-cycles positifs, mixtes et négatifs sont donnés les suivants, où $-(p|m)$ vaut 0 si p divise m et 1 sinon :*

	Double-cycles positifs $\mathcal{D}_{\ell,r}^{+,+}$	Double-cycles mixtes $\mathcal{D}_{\ell,r}^{-,+}$	Double-cycles négatifs $\mathcal{D}_{\ell,r}^{-,-}$
Ordre ω	Δ	r	$\begin{cases} \frac{\ell+r}{2} & \text{si } \frac{\ell+r}{\Delta} = 4 \\ \ell+r & \text{sinon} \end{cases}$
Nombre de configurations de période p , $p \omega$	$X^+(p)$	$X_{\ell}^{-,+}(p) = -(p \ell) \cdot L(\frac{p}{\Delta_p})^{\Delta_p}$	$X_{\Delta}^{-,-}(p) = -(p \Delta) \cdot P(\frac{p}{\Delta_p})^{\Delta_p}$
Nombre de configurations de période minimale p , $p \omega$	$\tilde{X}^+(p)$	$\tilde{X}_{\ell}^{-,+}(p) = \sum_{\substack{d p \\ -(d \ell)}} \mu(\frac{p}{d}) \cdot L(\frac{d}{\Delta_d})^{\Delta_d}$	$\tilde{X}_{\Delta}^{-,-}(p) = \sum_{\substack{d p \\ -(d \Delta)}} \mu(\frac{p}{d}) \cdot P(\frac{d}{\Delta_d})^{\Delta_d}$
Nombre d'attracteurs de période p , $p \omega$	$A^+(p)$	$A_{\ell}^{-,+}(p) = \frac{\tilde{X}_{\ell}^{-,+}(p)}{p}$	$A_{\Delta}^{-,-}(p) = \frac{\tilde{X}_{\Delta}^{-,-}(p)}{p}$
Nombre total d'attracteurs	$T^+(\omega)$	$T_{\ell}^{-,+}(\omega) = \frac{1}{r} \sum_{\substack{d r \\ -(d \ell)}} \phi(\frac{r}{d}) \cdot L(\frac{d}{\Delta_d})^{\Delta_d}$	$T_{\Delta}^{-,-}(\omega) = \frac{1}{n} \sum_{\substack{d n \\ -(d \Delta)}} \phi(\frac{n}{d}) \cdot P(\frac{d}{\Delta_d})^{\Delta_d}$

Entre autres choses (cf. figure 6), il ressort de ces résultats que (i) les DCAB positifs admettent deux configurations stables x et \bar{x}^V (avec $x = (0, \dots, 0)$ lorsqu'ils sont canoniques) et qu'ils ont un comportement asymptotique analogue aux CAB positifs de même ordre, (ii) les DCAB mixtes admettent une unique configuration stable, et (iii) les DCAB négatifs n'admettent aucune configuration stable.

Aussi, on remarque que, contrairement aux CAB, l'ordre des DCAB n'est pas nécessairement atteint. Enfin, ce théorème met là encore en exergue que, à l'instar des CAB, les DCAB admettent un nombre exponentiel d'attracteurs au regard de leur taille. Toutefois, malgré son caractère exponentiel, le nombre d'attracteurs des DCAB est significativement plus petit que celui des CAB. En d'autres termes, les intersections de cycles semblent participer fortement à la diminution des degrés de liberté asymptotiques des RI. En se fondant sur cette idée, des études ont été menées pour comparer $T^+(n)$ et $T^-(2n)$ des CAB aux quantités $T^+(\Delta)$, $T_{\ell}^{-,+}(r)$ et $T_{\Delta}^{-,-}(\ell+r)$ des DCAB.

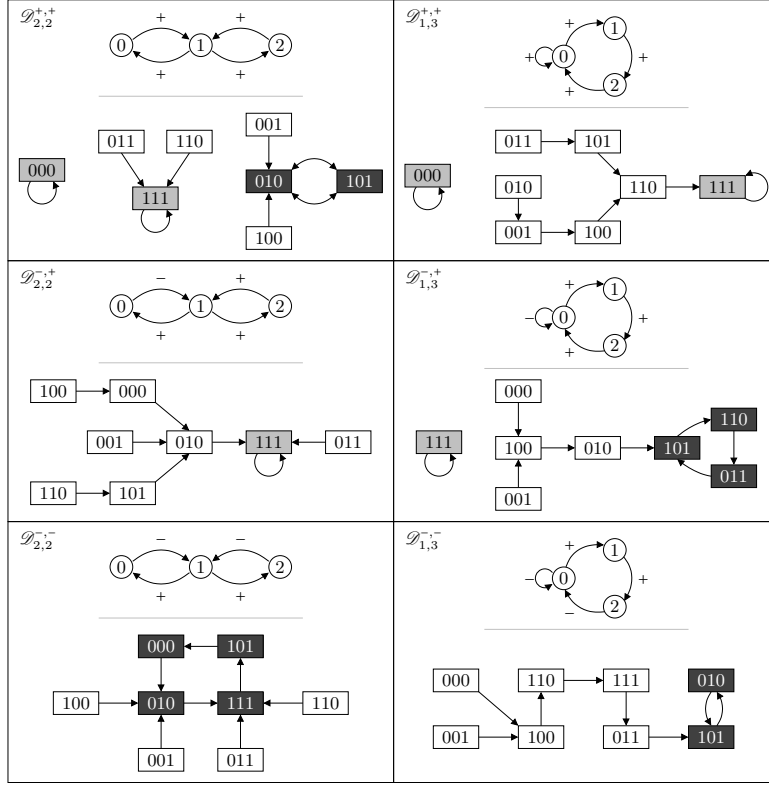


FIGURE 6 – Graphes d’interactions et de transitions de différents DCAB de taille $n = 3$ évoluant selon le mode de mise à jour parallèle, et où $\diamond = \vee$.

Considérons un réseau \mathcal{R} d’ordre ω , tel que $\mathcal{R} = \mathcal{C}_n^s$ ou $\mathcal{R} = \mathcal{D}_{\ell,r}^{s,s'}$, où $s, s' \in \{+, -\}$. Soit p un diviseur de ω . Notons également $\mathbb{Q} \in \{\mathbf{X}, \tilde{\mathbf{X}}, \mathbf{A}, \mathbf{T}\}$ l’une des quatre quantités analysées. Il a été démontré dans [40, 53] que $\mathbb{Q}_{\ell,r}^{+,+}(p) = \mathbb{Q}^+(p)$, $\mathbf{X}_{\ell,r}^{-,+}(p) \leq \mathbf{X}^+(p)$, $\mathbf{X}_{\ell,r}^{-,-}(p) \leq \mathbf{X}^+(p)$ et $\mathbb{Q}_{\ell,\ell}^{s,s}(p) = \mathbb{Q}_{\ell}^s(p)$.

Toutefois, le nombre de configurations récurrentes d’un DCAB, $\mathbf{X}_{\ell,r}^{s,s'}(p)$, a pu être borné plus finement en fonction de \mathbf{X}^+ et \mathbf{X}_n^- , en se fondant sur les résultats précédents ainsi que sur le lien de la suite de Lucas avec le *nombre d’or* [44], noté $\varrho = (1 + \sqrt{5})/2$ (racine de $x^2 - x - 1 = 0$) et celui de la suite de Perrin avec le *nombre plastique* [60], noté $\xi = \sqrt[3]{\frac{1}{2} + \frac{1}{6}\sqrt{\frac{23}{3}}} + \sqrt[3]{\frac{1}{2} - \frac{1}{6}\sqrt{\frac{23}{3}}}$ (racine de $x^3 - x - 1 = 0$). Les bornes trouvées ont amené au théorème 7, qui conclut cette section en mettant en évidence que la très grande majorité des configurations récurrentes ont la plus grande période minimale possible.

Théorème 7. *Soit \mathcal{R} un RAB d’ordre ω . Si \mathcal{R} est un CAB ou un DCAB tel que \mathcal{R} n’est ni $\mathcal{D}_{5,1}^{-,-}$ ni $\mathcal{D}_{1,5}^{-,-}$, alors son nombre total d’attracteurs $T(\omega)$ est*

borné par son nombre total de configurations récurrentes $X(\omega)$ tel que :

$$\frac{X(\omega)}{\omega} \leq T(\omega) \leq 2 \cdot \frac{X(\omega)}{\omega},$$

ce qui signifie que les périodes des attracteurs de \mathcal{R} sont très grandes :

$$\sum_{p|\omega} p \cdot \frac{A(p)}{T(\omega)} = \frac{X(\omega)}{T(\omega)} \geq \frac{\omega}{2}.$$

6.2 Double-cycles asynchrones

Nous abordons maintenant la dernière partie de cette synthèse, consacrée à la dynamique des DCAB asynchrones, en présentant la caractérisation récemment établie dans [36]. Au delà du fait qu'elle constitue le pendant de celle réalisée dans le contexte du mode de mise à jour parallèle, cette étude met en avant une approche nouvelle, fondée sur un formalisme mettant en œuvre des algorithmes pour décrire les longues séquences de mises à jour, qui permet une description élégante et plus fine de la dynamique des rétroactions dans les RI que les travaux précédents, en facilitant notamment l'étude des temps de convergence.

6.2.1 Définitions et notations

États et configurations – Ici, pour des raisons de clarté, nous utiliserons indépendamment la notation classique de l'ensemble V et sa congruence $V \equiv \{c = c_0, c_1, \dots, c_{n-1}\}$. Une configuration $x \in \mathbb{B}^n$ est vue comme un vecteur de deux mots binaires, dans lesquels le premier symbole représente $x_c \equiv x_0$. La configuration nulle est donc notée $(0^\ell, 0^r)$. Aussi, l'on note x^ℓ (resp. x^r) la projection de x sur le cycle \mathcal{C}_ℓ (resp. \mathcal{C}_r). Ainsi, $x = (x^\ell, x^r)$ et l'état de l'automate c_i^ℓ dans x est x_i^ℓ . Remarquons que $x_0 = x_0^\ell = x_0^r$ puisque les trois notations représentent l'état de l'automate c dans x .

Mesure d'expressivité – Soit x une configuration d'un CAB \mathcal{C}_n . Son *expressivité* est le nombre de facteurs 01 qui la composent, à savoir $|\{i \mid 0 \leq i \leq n-1, x_i = 0 \text{ et } x_{i+1 \bmod n} = 1\}|$. L'expressivité d'une configuration x d'un DCAB est la somme des expressivités de x^ℓ et x^r . De cette définition découle que, si ℓ et r sont pairs évidemment, les configurations les moins expressives sont $(0^\ell, 0^r)$ et $(1^\ell, 1^r)$ et que les plus expressives sont $((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}})$ et $((01)^{\frac{\ell}{2}}, (01)^{\frac{r}{2}})$.

Instructions élémentaires – Ce travail de caractérisation repose sur la mise en exergue de séquences spécifiques de mise à jour entre des configurations. Or, la longueur de ces séquences empêche d'en extraire directement l'essence. Il est proposé de voir ces séquences comme des *instructions* qui permettent d'en capturer plus simplement l'effet. Considérons donc \mathcal{D}_n un DCAB, \mathcal{C} de taille $\text{taille}(\mathcal{C})$ l'un de ses CAB, x la configuration courante de \mathcal{C} , et c_i et c_j

deux automates de \mathcal{C} distincts de c et tels que $i < j$, sur lesquels se fondent les sept instructions élémentaires suivantes :

- Mise à jour de c

$$\text{sync} : x_c \leftarrow f_c(x)$$

- Mise à jour de c_i

$$\text{update}(c_i) : x_{c_i} \leftarrow f_{c_i}(x)$$

- Mise à jour incrémentale

$$\text{incUp}(\mathcal{C}, i, j) : \text{pour } k = i \text{ à } j \text{ faire } \text{update}(c_k)$$

incUp admet la propriété suivante : soit x' le résultat de l'exécution de $\text{incUp}(\mathcal{C}, i, j)$ sur une configuration x . On a : $\forall k \in \{i, \dots, j\}, x'_k = x_{i-1}$ et $\forall k \notin \{i, \dots, j\}, x'_k = x_k$.

- Propagation incrémentale totale de x_c

$$\text{erase}(\mathcal{C}) : \text{incUp}(\mathcal{C}, 1, \text{taille}(\mathcal{C}) - 1)$$

erase admet la propriété suivante : soit x' le résultat de l'exécution de $\text{erase}(\mathcal{C})$ sur x . On a : $\forall k \in \{0, \dots, \text{taille}(\mathcal{C}) - 1\}, x'_k = x_0$.

- Propagation incrémentale de x_c sans perte d'expressivité

$$\text{expand}(\mathcal{C}) : \text{incUp}(\mathcal{C}, 1, \kappa - 1 \in \mathbb{N}) \text{ où}$$

$$\kappa = \min_{1 \leq k \leq \text{taille}(\mathcal{C}) - 1} \left\{ k \mid \begin{cases} (x_k = 0) \text{ et } (x_{k+1 \bmod \text{taille}(\mathcal{C})} = 1) & \text{si } x_c = 1 \\ (x_k = 1) \text{ et } (x_{k+1 \bmod \text{taille}(\mathcal{C})} = 0) & \text{si } x_c = 0 \end{cases} \right\}.$$

- Mise à jour décrémentele

$$\text{decUp}(\mathcal{C}, i, j) : \text{pour } k = j \text{ à } i \text{ faire } \text{update}(c_k)$$

decUp admet la propriété suivante : soit x' le résultat de l'exécution de $\text{decUp}(\mathcal{C}, i, j)$ sur x . On a : $\forall k \in \{i, \dots, j\}, x'_k = x_{k-1}$ et $\forall k \notin \{i, \dots, j\}, x'_k = x_k$.

- Mise à jour décrémentele totale (excepté c)

$$\text{shift}(\mathcal{C}) : \text{decUp}(\mathcal{C}, 1, \text{taille}(\mathcal{C}) - 1)$$

6.2.2 Résultats

Instructions plus complexes – Soit x une configuration d'un DCAB \mathcal{D} . Considérons un algorithme composé d'instructions définissant une séquence de mise à jour $\text{sequence}(x)$ à partir de x . Par abus de langage, $\text{sequence}(x)$ représente la séquence ainsi que la configuration résultant de son exécution. Pour les besoins de l'étude, on introduit trois autres séquences plus complexes dans le tableau 1. De plus, le lemme 1 suivant souligne que copy permet de transformer x en une configuration x' si x est suffisamment expressive.

Lemme 1. *Soit \mathcal{D} un DCAB et x et x' deux de ses configurations telles que $x_0 = x'_0$. Si, $\forall m \in \{\ell, r\}$, on a l'une des propriétés suivantes pour x :*

1. $\forall i \in \{1, \dots, \text{taille}(\mathcal{C}_m) - 1\}, x_i^m \neq x_{i-1}^m,$
2. $\forall i \in \{1, \dots, \text{taille}(\mathcal{C}_m) - 2\}, x_i^m \neq x_{i-1}^m$ et $x_{\text{taille}(\mathcal{C}_m) - 1}^m = x_{\text{taille}(\mathcal{C}_m) - 1}^m,$

<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">copy_c(x, x', \mathcal{C}_m)</div> 01. $\eta \leftarrow \text{taille}(\mathcal{C}_m)$; 02. si ($x_{\eta-1}^m = x_{\eta-2}^m$ et $x_{\eta-1}^m \neq x_{\eta-1}^m$) alors 03. $j \leftarrow \max\{k \mid k < \eta - 1 \text{ et } x_k^m \neq x_k^m\}$; 04. sinon $j \leftarrow \eta$; 05. fin si 06. pour ($k = \eta - 1$) à ↓ ($j + 1$) faire 07. $\text{update}(c_{k-1}^m)$; 08. $\text{update}(c_k^m)$; 09. fin faire 10. pour ($k = j - 1$) à ↓ (1) faire 11. si ($x_k^m \neq x_k^m$) alors $\text{update}(c_k^m)$; 12. finsi 13. fin faire	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">copy(x, x')</div> 01. copy_c(x, x', \mathcal{C}_ℓ) ; 02. copy_c(x, x', \mathcal{C}_r) ;
	<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">copy_p(x, x')</div> 01. si ($x_0 \neq x'_0$) alors 02. $\text{shift}(\mathcal{C}_\ell)$; 03. $\text{shift}(\mathcal{C}_r)$; 04. sync ; 05. fin si 06. copy(x, x') ;

TABLE 1 – Les instructions/séquences copy_c, copy et copy_p.

3. $\forall i \in \{1, \dots, \text{taille}(\mathcal{C}_m) - 2\}$, $x_i^m \neq x_{i-1}^m$ et $\exists p \in \{1, \dots, \text{taille}(\mathcal{C}_m) - 2\}$, $x_p^m \neq x_p^m$,

alors $\text{copy}(x, x') = x'$ et cette séquence exécute au plus $2(n + m - 6)$ mises à jour.

Ce lemme met en avant le pouvoir expressif des instructions/séquences pour capturer les trajectoires entre configurations. Concentrons-nous à présent sur les comportements des double-cycles à proprement parler, d'un point de vue général. Dans tous les algorithmes qui suivent, on considère qu'un DCAB \mathcal{D} est une variable globale.

DCAB *positifs* – Dans la section 6.1, on a vu que les DCAB positifs se comportent comme les CAB positifs, à savoir qu'ils admettent deux configurations stables parmi leurs attracteurs. Dans le cas asynchrone, ces deux configurations stables sont les uniques attracteurs. L'idée générale de la démonstration se fonde sur les DCAB positifs canoniques et établit que les deux séquences **fix0** et **fix1** données dans le tableau 2 permettent de transformer toute configuration avec au moins un automate à l'état 0 en $(0^\ell, 0^r)$, et toute configuration avec au moins un automate à l'état 1 dans chacun de ses CAB en $(1^\ell, 1^r)$, propriété illustrée dans le haut de la figure 7. Ce résultat est résumé dans le théorème 8 suivant.

Théorème 8. *Soit $\mathcal{D}_{\ell,r}^{+,+}$ un double-cycle positif canonique et x l'une de ses configurations instables. Si x admet un automate à l'état 0, alors $\text{fix0}(x) = (0^\ell, 0^r)$. De plus, si x admet un automate à l'état 1 dans chacun des cycles qui le compose, alors $\text{fix1}(x) = (1^\ell, 1^r)$. Le temps de convergence de $\mathcal{D}_{\ell,r}^{+,+}$ est au plus $2(n + m) - 5$.*

<div style="border: 1px solid black; display: inline-block; padding: 2px; margin-bottom: 5px;">fix0(x)</div> 01. si ($x_0 = 1$) alors 02. $i \leftarrow \min\{k \mid x_k^\ell = 0\}$; 03. incUp ($\mathcal{C}_\ell, i+1, n-1$) ; 04. sync ; 05. fin si 06. erase (\mathcal{C}_ℓ) ; 07. erase (\mathcal{C}_r) ;	<div style="border: 1px solid black; display: inline-block; padding: 2px; margin-bottom: 5px;">fix1(x)</div> 01. si ($x_0 = 0$) alors 02. $i \leftarrow \min\{k \mid x_k^\ell = 1\}$; 03. incUp ($\mathcal{C}_\ell, i+1, n-1$) ; 04. $j \leftarrow \min\{k \mid x_k^r = 1\}$; 05. incUp ($\mathcal{C}_r, j+1, m-1$) ; 06. sync ; 07. fin si 08. erase (\mathcal{C}_ℓ) ; 09. erase (\mathcal{C}_r) ;
--	---

TABLE 2 – Les séquences **fix0** et **fix1**.

DCAB *mixtes* – Pour ces double-cycles évoluant en asynchrone, à l’instar des DCAB positifs, l’asynchronisme permet de supprimer toutes les instabilités locales et fait que les DCAB mixtes n’admettent qu’un unique attracteur, qui est une configuration stable. Ce résultat met en avant, dans le cas canonique, la séquence **simp** présentée dans le tableau 3, qui fournit une façon de converger vers cette configuration stable depuis n’importe quelle configuration initiale x , en réduisant progressivement son expressivité. Cette propriété est elle-aussi illustrée dans la figure 7 et est formalisée par le théorème 9 suivant.

Théorème 9. *Soit $\mathcal{D}_{\ell,r}^{-,+}$ un DCAB mixte canonique, $\forall x \in \mathbb{B}^{\ell+r-1}$, $\mathbf{simp}(x) = (0^\ell, 0^r)$ et exécute au plus $2n + m - 2$ mises à jour.*

DCAB *négatifs* – Sont distingués ici les DCAB négatifs pairs, *i.e.* composés de deux CAB de taille paire, et impairs, *i.e.* composés d’au moins un CAB de taille impaire.

Les DCAB négatifs pairs admettent un unique attracteur, une oscillation stable de longueur $2^{\ell+r-1}$ induisant que toutes les configurations sont récurrentes et donc que le temps de convergence est nul. Toutefois, bien que toutes les configurations soient atteignables, les résultats révèlent que celles d’expressivité maximale sont difficiles à atteindre, dans le sens où le nombre de mises à jour pour les atteindre est quadratique en la taille du double-cycle. L’idée générale de la démonstration suit les trois points suivants (référencés respectivement par P1, P2 et P3 dans la suite) :

1. n’importe quelle configuration peut atteindre $(0^\ell, 0^r)$ en temps linéaire ;
2. $(0^\ell, 0^r)$ peut atteindre $((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}})$ en temps quadratique ;
3. toute configuration peut être atteinte depuis $((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}})$ en temps linéaire.

<div style="border: 1px solid black; display: inline-block; padding: 2px; margin-bottom: 5px;">$\text{simp}(x)$</div> 01. si $(x_0 = 1)$ alors 02. $\text{erase}(\mathcal{C}_\ell)$; 03. sync ; 04. fin si 05. $\text{erase}(\mathcal{C}_\ell)$; 06. $\text{erase}(\mathcal{C}_r)$;	<div style="border: 1px solid black; display: inline-block; padding: 2px; margin-bottom: 5px;">$\text{comp1}(x)$</div> 01. pour $(i = 1)$ à \uparrow $(\ell - 1)$ faire 02. sync ; 03. $\text{expand}(\mathcal{C}_\ell)$; 04. $\text{erase}(\mathcal{C}_r)$; 05. fin faire	<div style="border: 1px solid black; display: inline-block; padding: 2px; margin-bottom: 5px;">$\text{comp2}(x)$</div> 01. si $(x^r = 1^r)$ alors 02. sync ; 03. $\text{erase}(\mathcal{C}_r)$; 04. fin si 05. sync ; 06. $\text{expand}(\mathcal{C}_r)$; 07. pour $(i = 1)$ à \uparrow $(r - 2)$ faire 08. $\text{shift}(\mathcal{C}_\ell)$; 09. sync ; 10. $\text{expand}(\mathcal{C}_r)$; 11. fin faire
---	---	--

TABLE 3 – Les séquences simp , comp1 et comp2 .

Considérons P1. Il est facile de voir que la séquence simp demeure efficace pour atteindre $(0^\ell, 0^r)$, ce qui se formalise par le lemme 2 suivant.

Lemme 2. *Pour toute configuration x de $\mathcal{D}_{\ell,r}^{-,-}$, $\text{simp}(x) = (0^\ell, 0^r)$ et exécute au plus $2n + m - 2$ mises à jour.*

Considérons maintenant P2. Implicitement, ce point demande d'augmenter l'expressivité de $(0^\ell, 0^r)$ par des mises à jour successives, et donc de trouver une trajectoire jusqu'à $((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}})$. Pour ce faire, on procède en deux étapes, en augmentant l'expressivité de \mathcal{C}_ℓ en utilisant comp1 (cf. lemme 3) puis en augmentant celle de \mathcal{C}_r tout en ne réduisant pas celle de \mathcal{C}_ℓ en utilisant comp2 (cf. lemme 4). On obtient ensuite directement le résultat attendu, formalisé dans le lemme 5, par la composition $\text{comp} = \text{comp2} \circ \text{comp1}$.

Lemme 3. *Dans un DCAB négatif pair $\mathcal{D}_{\ell,r}^{-,-}$, $\text{comp1}((0^\ell, 0^r)) = ((10)^{\frac{n}{2}}, 1^m)$ et exécute au plus $(n - 1)(n + m - 2)$ mises à jour.*

Lemme 4. *Dans un DCAB négatif pair $\mathcal{D}_{\ell,r}^{-,-}$, $\text{comp2}(((10)^{\frac{n}{2}}, 1^m)) = ((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}})$ et exécute au plus $(m - 2)(n + m - 2) + (2m - 1)$ mise à jour.*

Lemme 5. *Dans un DCAB négatif pair $\mathcal{D}_{\ell,r}^{-,-}$, $\text{comp}((0^\ell, 0^r)) = ((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}})$ et exécute au plus $(n + m)^2 - 5(n - 1) - 3m$ mises à jour.*

P3 est quant à lui développé dans le lemme 6, qui utilise la séquence copy_p (cf. Table 1).

Lemme 6. *Dans un DCAB négatif pair $\mathcal{D}_{\ell,r}^{-,-}$, pour toute configuration x' , $\text{copy_p}(((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}}), x')$ transforme la configuration $((10)^{\frac{\ell}{2}}, (10)^{\frac{r}{2}})$ en x' en au plus $3(n + m - 4) - 1$ mises à jour.*

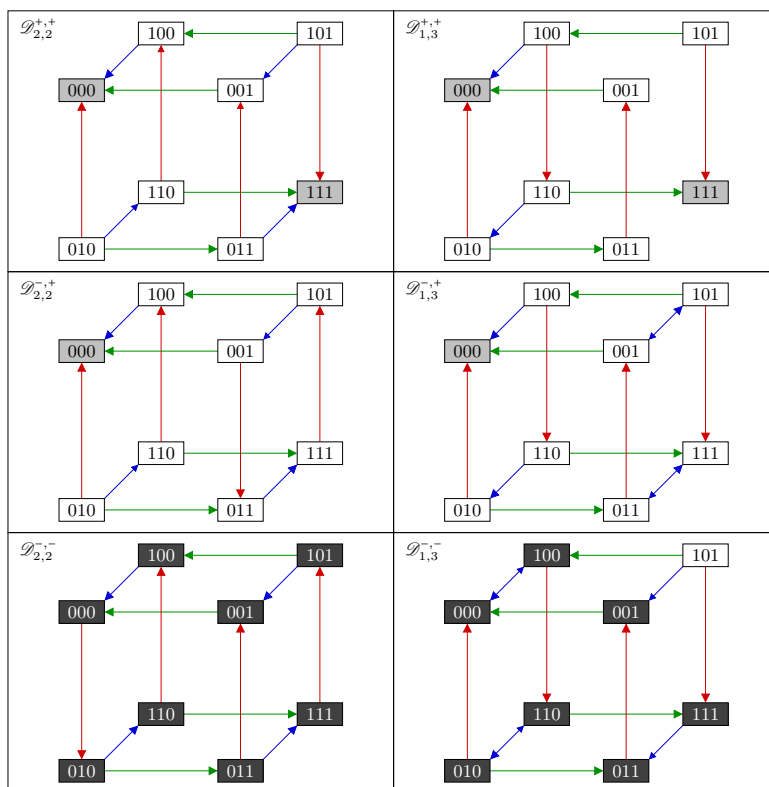


FIGURE 7 – Graphes de transitions asynchrones des différents DCAB de taille $n = 3$ présentés dans la figure 6, où $\diamond = \wedge$, en respectant le même ordre (les transitions inefficaces sont omises).

En partant des lemmes 2, 5 et 6, quelles que soient les configurations x et x' , la composition $\text{copy_p}(\text{comp}(\text{simp}(x)), x') = x'$ et démontre qu'il existe un unique attracteur de longueur $2^{\ell+r-1}$. D'où le théorème 10 suivant qui donne entre autres des bornes sur les temps de convergence (qui sont par ailleurs complétées et affinées dans [36]).

Théorème 10. *Un double-cycle négatif pair admet un unique attracteur de longueur $2^{\ell+r-1}$. Dans cette oscillation stable, toute configuration peut être atteinte depuis n'importe quelle autre en $O(n^2 + m^2)$ mises à jour. Toutefois, les configurations $(0^\ell, 0^r)$ et $(1^\ell, 1^r)$ peuvent être atteintes depuis n'importe quelle autre en $O(n + m)$ mises à jour, et les configurations $((01)_{\frac{\ell}{2}}, (01)_{\frac{r}{2}})$ et $((10)_{\frac{\ell}{2}}, (10)_{\frac{r}{2}})$ peuvent atteindre toutes les autres en $O(n + m)$ mises à jour.*

Tout comme les DCAB négatifs pairs, les DCAB négatifs impairs admettent également un unique attracteur, une oscillation stable en l'occurrence. Toutefois, toutes les configurations de ces derniers ne sont pas nécessairement récurrentes. En effet, ils admettent un ensemble I de configurations non

atteignables, dites irréversibles. La démonstration du résultat principal qui généralise par ailleurs le théorème 10 dont les bornes restent valides, et qui est formalisé dans le théorème 11 suivant, repose sur la caractérisation de I .

Théorème 11. *Soit $\alpha : \mathbb{N} \rightarrow \{0, 1\}$, avec $\alpha(k) = \begin{cases} 0 & \text{si } k = 0 \text{ ou } k \equiv 1 \pmod{2} \\ 1 & \text{sinon} \end{cases}$.
 Tout DCAB $\mathcal{D}_{\ell,r}^{-,-}$ admet un unique attracteur $\mathbb{B}^{\ell+r-1} \setminus I$, où $|I| = \alpha(\ell - 1) \times 2^{r-1} + \alpha(r - 1) \times 2^{\ell-1}$.*

7 Conclusion

Dans cet article, nous avons synthétisé les résultats majeurs obtenus ces dernières années sur le rôle des rétroactions dans les réseaux d'interactions. Plus précisément, en raison de leur caractère essentiel avéré par la littérature, nous avons volontairement mis l'accent sur la dynamique des cycles et double-cycles, en nous intéressant particulièrement à l'influence que jouent sur eux les modes de mise à jour. Sans revenir sur les résultats eux-mêmes, cet article met en exergue les différences fondamentales induites par l'« ordonnancement » des événements sur les comportements des moteurs de complexité comportementale des réseaux d'interactions : là où le parallélisme tend à rendre toute la richesse asymptotique mathématique des réseaux, l'asynchronisme pur tend quant à lui à réduire les degrés de liberté des réseaux, ce qui explique en partie pourquoi le premier est souvent préféré dans les travaux théoriques alors que le second est plébiscité dans les travaux qui sont orientés vers les applications à la biologie moléculaire. À l'heure actuelle, aucune connaissance réelle en biologie moléculaire n'établit précisément la manière dont les régulations s'exécutent dans le temps. Le type d'études fondamentales développé dans cette synthèse, axé sur les modes de mise à jour, reste donc essentiel aujourd'hui pour aller plus loin dans la compréhension formelle et appliquée des réseaux d'interactions. Il convient donc de poursuivre dans ce sens.

Une autre ouverture naturelle mise en avant par ces travaux, très ancrée en informatique fondamentale, consiste à développer plus avant les connaissances sur les propriétés dynamiques des réseaux d'interactions, et notamment des relations entre leur architecture et leur structure. Mais d'autres pistes tout aussi pertinentes s'ouvrent également. L'une d'elles a été soulevée à propos des double-cycles asynchrones et traite de la complexité en temps des réseaux. Une question qui reste actuellement ouverte est la suivante : la complexité en temps des réseaux va-t-elle de paire avec leur richesse comportementale ? Si oui, peut-on en trouver une mesure ? Enfin, les travaux menés sur les cycles et double-cycles en parallèle, développés dans [40, 53], évoquent des perspectives autour de la calculabilité, de la modularité, de l'universalité intrinsèque des réseaux, qui sont certainement, à l'instar de la

compréhension toujours plus fine des influences des modes de mise à jour, parmi les pistes les plus prometteuses du domaine.

Remerciements Le travail présenté dans cet article a été en partie financé par les projets ANR *Synbiotic* et *QuasiCool*, ainsi que par le projet RNSC *Météding* et le projet IXXI *Maajes*.

Références

- [1] M. Aldana. Boolean dynamics of networks with scale-free topology. *Physica D*, 185 :45–66, 2003.
- [2] T. M. Apostol. *Introduction to analytic number theory*. Springer-Verlag, 1976.
- [3] J. Aracena, M. González, A. Zuñiga, M. A. Mendez, and V. Cambiazo. Regulatory network for cell shape changes during drosophila ventral furrow formation. *Journal of Theoretical Biology*, 239 :49–62, 2006.
- [4] E. R. Berlekamp, J. H. Conway, and R. K. Guy. *Winning ways for your mathematical plays*. Academic Press, 1982.
- [5] J. Berstel and D. Perrin. The origins of combinatorics on words. *European Journal of Combinatorics*, 28 :996–1022, 2007.
- [6] A. Church. A set of postulates for the foundation of logic. *Annals of Mathematics*, 33 :346–366, 1932.
- [7] M. Cook. Universality in elementary cellular automata. *Complex Systems*, 15 :1–40, 2004.
- [8] M. Cosnard and J. Demongeot. On the definitions of attractors. In *Iteration theory and its functional equations*, volume 1163 of *Lecture Notes in Mathematics*, pages 23–31. Springer, 1985.
- [9] M. Cosnard, J. Demongeot, and A. Le Breton, editors. *Rhythms in biology and other fields of application*, volume 49 of *Lecture Notes in Biomathematics*. Springer, 1983.
- [10] P. Cull. Linear analysis of switching nets. *Biological Cybernetics*, 8 :31–39, 1971.
- [11] M. Delbrück. Génétique du bactériophage. In *Unités biologiques douées de continuité génétique*, volume 8 of *Colloques internationaux du CNRS*, pages 91–103, 1949.
- [12] J. Demongeot. *Au sujet de quelques modèles stochastiques appliqués à la biologie*. PhD thesis, Université scientifique et médicale de Grenoble, 1975.
- [13] J. Demongeot, E. Goles, and M. Tchuente, editors. *Dynamical systems and cellular automata*. Academic Press, 1985.

- [14] J. Demongeot, M. Noulal, and S. Sené. Combinatorics of Boolean automata circuits dynamics. *Discrete Applied Mathematics*, 160 :398–415, 2012.
- [15] B. Elspas. The theory of autonomous linear sequential networks. *IRE Transactions on Circuit Theory*, 6 :45–60, 1959.
- [16] C. Gershenson. Updating schemes in random Boolean networks: do they really matter? In *Proceedings of Artificial Life*, pages 238–243. MIT Press, 2004.
- [17] E. Goles. Fixed point behavior of threshold functions on a finite set. *SIAM Journal on Algebraic and Discrete Methods*, 3 :529–531, 1982.
- [18] E. Goles, F. Fogelman-Soulié, and D. Pellegrin. Decreasing energy functions as a tool for studying threshold networks. *Discrete Applied Mathematics*, 12 :261–277, 1985.
- [19] E. Goles and S. Martínez. *Neural and automata networks: dynamical behavior and applications*, volume 58 of *Mathematics and Its Applications*. Kluwer Academic Publishers, 1990.
- [20] S. W. Golomb. *Shift register sequences*. Holden-Day Inc., 1967.
- [21] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete mathematics: a foundation for computer science*. Addison-Wesley, 1989.
- [22] K. F. Gödel. *Kurt Gödel collected works*, volume I – Publications 1929–1936, chapter On undecidable propositions of formal mathematical systems, pages 346–372. Oxford University Press, 1986.
- [23] B. Hesper and P. Hogeweg. Bioinformatica: een werkconcept. *Kameleon*, 1 :18–29, 1970.
- [24] P. Hogeweg and B. Hesper. Interactive instruction on population interactions. *Computers in Biology and Medicine*, 8 :319–327, 1978.
- [25] D. A. Huffman. Canonical forms for information-lossless finite-state logical machines. *IRE Transactions on Information Theory*, 5 :41–59, 1959.
- [26] F. Jacob and J. Monod. Genetic regulatory mechanisms in the synthesis of proteins. *Journal of Molecular Biology*, 3 :318–356, 1961.
- [27] S. A. Kauffman. Homeostasis and differentiation in random genetic control networks. *Nature*, 224 :177–178, 1969.
- [28] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22 :437–467, 1969.
- [29] S. A. Kauffman. *Current topics in developmental biology*, volume 6, chapter Gene regulation networks: A theory for their global structures and behaviors, pages 145–181. Elsevier, 1971.

- [30] S. A. Kauffman, C. Peterson, B. Samuelsson, and C. Troein. Random Boolean network models and the yeast transcriptional network. *Proceedings of the National Academy of Sciences of the USA*, 100 :14796–14799, 2003.
- [31] M. Kaufman and R. Thomas. Towards a logical analysis of the immune response. *Journal of Theoretical Biology*, 114 :527–561, 1985.
- [32] S. C. Kleene. *Automata studies*, volume 34 of *Annals of Mathematics Studies*, chapter Representation of events in nerve nets and finite automata, pages 3–41. Princeton University Press, 1956.
- [33] H. T. Kung and C. E. Leiserson. *Introduction to VLSI systems*, chapter Algorithms for VLSI processor arrays, pages 271–292. Addison-Wesley, 1980.
- [34] P. Kurka. Languages, equicontinuity and attractors in cellular automata. *Ergodic Theory and Dynamical Systems*, 17 :417–433, 1997.
- [35] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Journal of Mathematical Biophysics*, 5 :115–133, 1943.
- [36] T. Melliti, M. Noul, D. Regnault, S. Sené, and J. Sobieraj. Full characterisation of attractors of two intersected asynchronous Boolean automata cycles. Technical report, Université d’Évry – Val d’Essonne, 2013. arXiv :1310.5747.
- [37] L. Mendoza and E. R. Alvarez-Buylla. Dynamics of the genetic regulatory network for *Arabidopsis thaliana* flower morphogenesis. *Journal of Theoretical Biology*, 193 :307–319, 1998.
- [38] J. Monod, J.-P. Changeux, and F. Jacob. Allosteric proteins and cellular control systems. *Journal of Molecular Biology*, 6 :306–329, 1963.
- [39] M. Noul. Dynamics of circuits and intersection circuits. In *Proceedings of LATA*, volume 7183 of *Lecture Notes in Computer Science*, pages 433–444. Springer, 2012.
- [40] M. Noul. *Updating automata networks*. PhD thesis, École normale supérieure de Lyon, 2012. <http://tel.archives-ouvertes.fr/tel-00726560>.
- [41] Y. Puri and T. Ward. Arithmetic and growth of periodic orbits. *Journal of Integer Sequences*, 4 :01.2.1, 2001.
- [42] É. Remy, B. Mossé, C. Chaouiya, and D. Thieffry. A description of dynamical graphs associated to elementary regulatory circuits. *Bioinformatics*, 19 :ii172–ii178, 2003.
- [43] É. Remy, P. Ruet, and D. Thieffry. Graphic requirements for multistability and attractive cycles in a Boolean dynamical framework. *Advances in Applied Mathematics*, 41 :335–350, 2008.

- [44] P. Ribenboim. *The new book of prime number records*. Springer-Verlag, 1996.
- [45] A. Richard. Negative circuits and sustained oscillations in asynchronous automata networks. *Advances in Applied Mathematics*, 44 :378–392, 2010.
- [46] A. Richard and J.-P. Comet. Necessary conditions for multistationarity in discrete dynamical systems. *Discrete Applied Mathematics*, 155 :2403–2413, 2007.
- [47] A. Richard, J.-P. Comet, and G. Bernot. R. Thomas’ modeling of biological regulatory networks : introduction of singular states in the qualitative dynamics. *Fundamenta Informaticae*, 65 :373–392, 2004.
- [48] F. Robert. Blocs-H-matrices et convergence des méthodes itératives classiques par blocs. *Linear Algebra and its Applications*, 2 :223–265, 1969.
- [49] F. Robert. Contraction en norme vectorielle : convergence d’itérations chaotiques pour des équations non linéaires de point fixe à plusieurs variables. *Linear Algebra and its Applications*, 13 :19–35, 1976.
- [50] F. Robert. *Discrete iterations : a metric study*, volume 6 of *Springer Series in Computational Mathematics*. Springer, 1986.
- [51] F. Robert. *Les systèmes dynamiques discrets*, volume 19 of *Mathématiques & Applications*. Springer, 1995.
- [52] F. Ruskey. Combinatorial generation. Book preliminary working draft, 2003.
- [53] S. Sené. *Sur la bio-informatique des réseaux d’automates*. PhD thesis, Université d’Évry – Val d’Essonne, 2012. <http://tel.archives-ouvertes.fr/tel-00759287>.
- [54] D. Thieffry and R. Thomas. Dynamical behaviour of biological regulatory networks – II. Immunity control in bacteriophage lambda. *Bulletin of Mathematical Biology*, 57 :277–297, 1995.
- [55] R. Thomas. Boolean formalization of genetic control circuits. *Journal of Theoretical Biology*, 42 :563–585, 1973.
- [56] R. Thomas. On the relation between the logical structure of systems and their ability to generate multiple steady states or sustained oscillations. In *Numerical methods in the study of critical phenomena*, volume 9 of *Springer Series in Synergetics*, pages 180–193. Springer, 1981.
- [57] R. Thomas. Regulatory networks seen as asynchronous automata : a logical description. *Journal of Theoretical Biology*, 153 :1–23, 1991.
- [58] R. Thomas, D. Thieffry, and M. Kaufman. Dynamical behaviour of biological regulatory networks – I. Biological role of feedback loops and practical use of the concept of the loop- characteristic state. *Bulletin of Mathematical Biology*, 57 :247–276, 1995.

- [59] A. M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 2 :230–265, 1936.
- [60] H. van der Laan. *Le nombre plastique, quinze leçons sur l'ordonnance architectonique*. E.J. Brill, 1960.
- [61] J. von Neumann. *Theory of self-reproducing automata*. University of Illinois Press, 1966.
- [62] S. Wolfram. Universality and complexity in cellular automata. *Physica D*, 10 :1–35, 1984.