

numéro

18

1024

B U L L E T I N

de la société informatique
de France

Novembre

2021

COMITÉ DE RÉDACTION

SYLVIE ALAYRANGUES
Université de Poitiers

OLIVIER BAUDON
Université de Bordeaux

YVES BERTRAND
Université de Poitiers

JEAN-PAUL DELAHAYE
Université de Lille

ISABELLE DEBLED-RENNESON
Université de Lorraine

MARIE DUFLLOT-KREMER
Université de Lorraine

THOMAS FERNIQUE
CNRS, Université Paris 13

PHILIPPE MARQUET
Université de Lille

EUNIKA MERCIER-LAURENT
Université de Lyon

PATRICE NAUDIN
Université de Poitiers

PIERRE PARADINAS
CNAM Paris

NICOLAS PASSAT
*Université de Reims
Champagne-Ardenne*

MARIA POTOP-BUTUCARU
Sorbonne Université

MICHEL RAYNAL
Université de Rennes

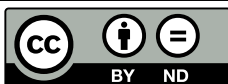
NATHALIE REVOL
Inria, Université de Lyon

NANCY RODRIGUEZ
Université de Montpellier

FLORENCE SÈDES
Université de Toulouse

DENIS PALLEZ, *Université Côte d'Azur, rédacteur en chef*

Contact : 1024@societe-informatique-de-france.fr



*Cette œuvre est mise à disposition sous licence Attribution - Pas de Modification 4.0 France.
Pour voir une copie de cette licence, visitez*

<http://creativecommons.org/licenses/by-nd/4.0/deed.fr>

*ou écrivez à Creative Commons, 444 Castro Street, Suite 900, Mountain View, California,
94041, USA.*

SOCIÉTÉ INFORMATIQUE DE FRANCE

Institut Henri Poincaré, 11 rue Pierre et Marie Curie, 75231 Paris Cedex 05

Prix public : 32 € (adhérents SIF : -30%)

Directeur de la publication : Yves Bertrand

ISSN : 2270-1419

Couverture : d'après une maquette réalisée par Lollygraph.com.

SOMMAIRE DU N° 18



SIF

Le mot du (nouveau) président, <i>Yves Bertrand</i>	3
---	---

JOURNÉE « REPRODUCTIBILITÉ DE LA RECHERCHE »

Introduction à la journée, <i>Damien Magoni</i>	5
La reproductibilité des calculs coûteux, <i>Konrad Hinszen</i>	11
Reproduire les environnements logiciels : un maillon incontournable de la recherche reproductible, <i>Ludovic Courtès</i>	15

DOSSIER INFORMATIQUE QUANTIQUE

L'ordinateur quantique : un objectif impossible à atteindre ? <i>Michel Dyakonov</i>	23
Les ordinateurs quantiques : un enthousiasme justifié, <i>Giuseppe Di Molfetta</i>	35
Écosystème quantique en France et dans le monde, <i>Olivier Ezratty</i>	43

DU CÔTÉ DES ASSOCIATIONS

FBP	59
MMI	63

DU CÔTÉ DES LABORATOIRES

GDR IG-RV	65
-----------------	----

ENSEIGNEMENT

La numérisation de l'école, fausse bonne idée du siècle, <i>Philippe Bihouix</i>	67
--	----

FEMMES ET INFORMATIQUE

Mendeleïeva : un jeu de plateau pour mettre en valeur les femmes en sciences, <i>Florence Sèdes</i>	81
« Les décodeuses du numérique », une BD qui parle de, et fait parler de #femmesInformatique, <i>Florence Sèdes</i>	85
La cordée de la réussite « décodeuses d'informatique », <i>Julien Narboux, Cristel Pelsser, Basile Sauvage et Gaëlle Thomas</i>	87

HISTOIRE

L'informatique de Claude Pair, <i>Marion Créhange, Pierre Lescanne et Alain Quéré</i>	91
Concours de programmes AFCET — 1981, <i>Jacques Baudé</i>	117

HOMMAGE

Hommage à Philippe Aigrain, <i>François Pellegrini</i>	129
Hommage à Paul Camion, <i>Anne Canteaut, Louis Goubin, Jacques Patarin</i>	133
Hommage à Claudine Hermann, <i>Florence Sèdes</i>	135
Sophia Antipolis : le père fondateur s'en est allé, <i>Gérard Giraudon</i>	137

NOTES DE LECTURE

Turing à la plage — L'intelligence artificielle dans un transat, <i>lu par Nathalie Revol</i>	141
Voyage au coeur de l'informatique : technologies, usages, enjeux, <i>lu par Michel Raynal</i>	145

PANDÉMATIQUE

Une nouvelle approche au traçage numérique des contacts (application COVI au Québec), <i>Tegan Maharaj et Yoshua Bengio</i>	147
---	-----

RÉCRÉATION

L'infection du damier, <i>Jean-Paul Delahaye</i>	161
--	-----



Le mot du (nouveau) président

Yves Bertrand¹

Chères adhérentes, chers adhérents,

La sortie du numéro 18 de notre bulletin semestriel coïncide, nous l'espérons tous, avec une autre esquisse de sortie : celle de la crise sanitaire qui bouleverse nos vies depuis plus d'un an et demi, en causant des millions de malades et de décès. Elle a véhiculé des débats, polémiques, contrevérités, et comportements irrationnels sans réel précédent, avec les réseaux sociaux comme caisse de résonance principale. Actuellement, l'intérêt de la vaccination fait quotidiennement débat dans notre pays en particulier. C'est aussi la caractéristique commune des six grands sujets abordés par ce bulletin : ils font ou peuvent faire débat. La reproductibilité en science, et en informatique notamment, est une préoccupation suffisamment récente pour que ses fondements mêmes, ses attendus et les pratiques correspondantes, soient le creuset d'approches variées et multidisciplinaires, comme en a attesté le succès de la journée « Reproductibilité » organisée par la SIF le 10 mai 2021. L'émergence d'une informatique quantique est suffisamment sujette à controverse pour que vous trouviez dans ce bulletin un article qui l'assimile à une chimère, un autre qui en brosse un tableau plein d'espoirs, et un troisième qui propose un état des lieux mondial des technologies et « marchés » sous-jacents. L'article sur la numérisation de l'école affiche son parti pris dès son titre : c'est une « fausse bonne idée », à l'heure où l'utilisation de l'outil numérique en formation est pour beaucoup synonyme de progrès pédagogique. La place des femmes dans l'informatique reste plus d'actualité que jamais, comme en témoigne le taux de filles très bas (13 %) dans la spécialité NSI en première au lycée, et plus généralement dans les études supérieures et les métiers du numérique. La variété des moyens pour faire croître ce taux est à la hauteur de

1. Président de la Société informatique de France, professeur des universités, université de Poitiers.

l'enjeu : l'approche ludique et historique du jeu « Mendeleïva », l'approche communicationnelle de la BD du CNRS « Les décodeuses du numérique » et l'approche « Cordées de la réussite » illustrent cette variété dans le présent bulletin. Les débats sur l'efficacité de chacune des approches affleurent et ceux sur la nécessité d'une approche radicale ou d'une approche plus progressive des questions d'égalité et de parité sont omniprésents. L'histoire même de l'informatique intègre autant d'approches (matérielle, sociologique, économique, scientifique, etc.) qu'elle compte de figures de proue : Marion Créhange, Pierre Lescanne et Alain Quéré reviennent sur les 18 premières années du parcours de l'un des fondateurs de l'informatique en France, avec une rédaction — aussi laudatrice qu'elle est méritée et aussi subjective qu'elle est revendiquée — de l'apport à notre discipline de l'un de ses plus éminents représentants : Claude Pair. Enfin, même si la pandémie faiblit à l'heure où ces lignes sont écrites, elle demeure suffisamment prégnante pour faire l'objet de recherches en traçage numérique. Elle apparaît même en filigrane dans la récréation de Jean-Paul Delahaye, et notre actualité nationale récente atteste à quel point elle fut et reste l'objet des débats les plus échevelés ! Le débat, voire la controverse, sont donc au cœur des thèmes abordés par ce 18^e numéro de 1024. Le rôle de la SIF est de promouvoir les formations, la recherche et les métiers de l'informatique et du numérique, de développer le plus possible toute forme de médiation qui concourra à leur promotion. La SIF doit également participer aux débats de la société dans laquelle notre discipline s'incarne, tant elle en bouleverse depuis un demi-siècle des pans entiers. C'est plus vrai encore à quelques mois des 10 ans de notre association, et à quelques mois de l'élection présidentielle. Cette participation nécessite que nous trouvions un équilibre délicat entre une position trop en retrait par rapport aux préoccupations sociétales et une position trop marquée politiquement, entre une approche purement savante et désincarnée et une posture trop polémique ou clivante. Quelques valeurs fondamentales que peuvent partager le plus grand nombre d'entre nous pourraient à court terme être portées à la connaissance de l'ensemble des femmes et des hommes qui s'estiment aujourd'hui en mesure de gouverner la France de 2022 à 2027 : une approche rationnelle de notre discipline, une approche humaniste, égalitaire, ouverte au monde, intégrant les préoccupations des ressources, du climat et de la santé... Gageons que la préparation du congrès des 10 ans de la SIF sera l'occasion de formaliser ensemble nos valeurs et nos aspirations pour prétendre peser, même modestement, sur les débats innombrables dans lesquels notre discipline et les femmes et les hommes qui l'incarnent sont engagés.



Introduction à la journée

Damien Magoni¹

La Société informatique de France (SIF) a organisé le 10 mai 2021 une journée de présentations et d'échanges sur la reproductibilité de la recherche², un thème qui a pris de plus en plus d'ampleur depuis une dizaine d'années. La journée a réuni huit orateurs sur des thèmes centrés autour de cette problématique. Cette journée s'est tenue en visio-conférence et les interventions ont été enregistrées³.

Après le mot de bienvenue d'Yves Bertrand, président de la SIF, un premier exposé introductif intitulé « La recherche reproductible : c'est quoi ? pourquoi en faire ? comment ? » a été donné par Christophe Pouzat (université de Strasbourg/CNRS). Sa présentation a commencé par un court historique des pratiques que désigne la « recherche reproductible ». Christophe a ensuite présenté certains des outils qui lui ont permis de la pratiquer concrètement depuis 2005. Il a enfin exposé les gains considérables que cette pratique permet de réaliser dans le cadre de projets impliquant des chercheurs de plusieurs disciplines. Chercheur CNRS à l'IRMA de Strasbourg, Christophe Pouzat est neurophysiologiste et travaille sur l'analyse de données.

L'exposé suivant intitulé « Archiver et référencer le code source des logiciels de recherche pour la science ouverte et la reproductibilité » a été donné par Roberto Di Cosmo (université Paris Diderot/Inria) et a concerné l'archive Software Heritage. Le logiciel est en effet un pilier fondamental de la recherche scientifique moderne, et il est primordial d'archiver et de référencer correctement les codes sources du logiciel utilisé dans toutes les activités de recherche. Dans sa présentation, Roberto Di Cosmo

1. Professeur en informatique, université de Bordeaux.

2. Les supports des présentations sont téléchargeables depuis la page web de l'évènement <https://www.societe-informatique-de-france.fr/journee-reproductibilite>.

3. Les vidéos sont accessibles via <https://vimeo.com/showcase/8652937>.

a montré comment utiliser l'infrastructure Software Heritage pour archiver de façon transparente tous les codes sources des logiciels et comment obtenir et utiliser les identifiants intrinsèques, spécifiquement conçus pour référencer le code source d'une manière qui facilite la reproductibilité à long terme. Roberto Di Cosmo est professeur d'informatique à l'université Paris Diderot, membre du comité d'orientation pour la science ouverte en France, et actuellement détaché chez Inria. Suivant de près l'impact de l'informatique sur la société, il prône depuis longtemps l'adoption du logiciel libre. En 2015, il a été à l'origine de Software Heritage, une initiative qui vise à construire l'archive universelle de tout le code source publiquement disponible, en partenariat avec l'UNESCO.

Konrad Hinsén du centre de biophysique moléculaire d'Orléans, nous a ensuite présenté ses travaux sur « La reproductibilité des calculs coûteux » qui indique comment assurer ou vérifier la reproductibilité d'un calcul qui occupe un centre de calcul pendant trois semaines ? Ou encore, comment faire si des contraintes techniques nous obligent à exécuter chaque étape d'un calcul sur un autre ordinateur ? Les outils standards pour la recherche reproductible n'ayant pas encore de réponses à ces problématiques, Konrad a indiqué comment le projet ActivePapers a tenté de relever ces défis, et avec quel degré de succès. Konrad Hinsén, travaille sur la structure et la dynamique des protéines avec les méthodes de la simulation moléculaire. Auteur de plusieurs bibliothèques scientifiques en Python, membre fondateur du projet NumPy et enseignant pour Software Carpentry, Konrad est co-auteur d'un MOOC sur la recherche reproductible avec Arnaud Legrand.

Le dernier exposé de la matinée a été consacré à « la randomisation, une solution aux difficultés de reproductibilité de mesures de performances de processeurs modernes ? » et a été présenté par Arnaud Legrand (université de Grenoble-Alpes/CNRS). Lorsque l'on parle de problèmes de reproductibilité de la recherche, on met souvent en avant des problèmes de provenance, de manque de contrôle du logiciel et de son environnement, de problèmes de stabilité numérique, etc. liés à un mauvais usage des statistiques. Les aspects mesures et expériences sont souvent assez peu évoqués, peut-être parce que les difficultés expérimentales sont un problème ancien et déjà assez bien cerné pour un certain nombre de disciplines (physique, biologie), ce qui n'est pas forcément le cas en informatique qui est une science relativement jeune. Les processeurs modernes sont des objets complexes avec des hiérarchies de caches au comportement parfois opaque, aux stratégies de vectorisation et de réordonnement dynamique d'instruction obscures, dont la fréquence de fonctionnement varie au cours du temps pour s'adapter à la charge de travail et à la température du processeur, etc. Effectuer des mesures fiables sur ces processeurs peut vite devenir assez difficile, et obtenir des comportements « similaires » entre deux machines supposées identiques, voire sur la même machine à deux périodes différentes, peut être un véritable casse-tête. L'exposé a présenté des plans d'expériences robustes à adopter permettant de se prémunir contre ces problèmes ou de les

identifier. Arnaud a aussi présenté quelques « histoires d'horreur » et les limitations potentielles de la randomisation dans ce contexte. Chercheur au CNRS depuis 2004, Arnaud Legrand s'intéresse à l'étude des infrastructures informatiques distribuées, à grande échelle, utilisées pour le calcul scientifique. Cela recouvre les questions d'optimisation liées à l'exploitation de ces plateformes (ordonnancement, optimisation combinatoire, théorie des jeux) et les techniques d'évaluation de performances (simulation, visualisation, analyse statistique). Il est l'un des concepteurs du projet SimGrid, un outil de simulation open source dont le but est de faciliter la recherche dans le domaine de l'optimisation des systèmes parallèles et distribués. Depuis 5 ans, il participe à la promotion de meilleures pratiques expérimentales et à l'amélioration de la méthodologie scientifique à travers des tutoriels, des conférences, des cours, et un MOOC sur la recherche reproductible avec Konrad Hinsin.

L'après-midi a été composé de quatre exposés dont le premier intitulé « Reproductibilité computationnelle en sciences de la vie et workflows scientifiques : état des lieux et retour d'expérience » a été présenté par Sarah Cohen-Boulakia (université Paris-Saclay). Sa présentation a dressé le bilan des travaux du groupe de travail ReproVirtuFlow du GDR MaDICS qui s'intéresse à la reproductibilité des analyses de données bio-informatiques. De nombreux outils et familles de solutions existent aujourd'hui pour tendre vers une meilleure reproductibilité computationnelle des résultats bio-informatiques. Son équipe a testé ces outils dans le cadre de l'organisation de « reprohackathons », hackathons dont l'objectif est de reproduire un résultat publié. Son équipe a proposé notamment la définition de différents niveaux de reproductibilité d'une analyse et l'identification de caractéristiques clés des outils existants vis-à-vis de leur capacité à reproduire des résultats. Professeur à l'université Paris-Saclay dans l'équipe bio-informatique du laboratoire interdisciplinaire des sciences du numérique (LISN), Sarah Cohen-Boulakia est directrice du GDR MaDICS. Son domaine d'expertise porte sur la reproductibilité des analyses bio-informatiques et en particulier sur la provenance dans les workflows scientifiques.

L'exposé suivant, intitulé « Archiver, identifier, décrire et citer le code source : le dépôt de logiciel de recherche sur l'archive ouverte HAL », a été présenté par Morane Gruenpeter (Software Heritage). L'importance de déposer et partager nos logiciels de recherche nécessite de savoir comment le faire. Morane a présenté quatre types d'utilisation indispensables pour déchiffrer le dépôt logiciel et les bonnes pratiques associées à chacun. Le premier usage concerne l'archivage du code source dans HAL et sur Software Heritage, qui est une action nécessaire pour assurer l'accessibilité à long terme. Le deuxième usage concerne l'identification de la version spécifique avec un identifiant intrinsèque, une première étape pour conquérir le défi de la reproductibilité. Le troisième usage concerne la description du code source pour une meilleure compréhension du logiciel et pour permettre la découverte du logiciel sur des moteurs de recherche. Et enfin, le dernier concerne la citation du logiciel pour attribuer le crédit aux auteurs dans l'écosystème académique. Morane a

rejoint l'équipe Software Heritage en 2017. En 2018-2019, elle a continué son travail de recherche en collaboration avec le projet européen EU2020 CROSSMINER pour construire le Web sémantique des projets FOSS par la compilation des métadonnées existantes. Elle est membre active de plusieurs groupes de travail dans les domaines de la citation du logiciel, de la préservation numérique et du Web sémantique.

Nicolas Rougier (Inria Bordeaux) a ensuite présenté ReScience C à travers son exposé intitulé *Long term reproducibility*. ReScience C est un journal à comité de lecture en accès ouvert qui cible la recherche computationnelle et encourage la réplification de travaux de recherche déjà publiés. Ce journal a été co-fondé par Konrad Hinsén et lui-même. Il promeut des nouvelles implémentations open-source afin de s'assurer que la recherche initiale soit reproductible. Son comité éditorial a organisé le *Ten Years Reproducibility Challenge* qui consiste à inviter les chercheurs à essayer d'exécuter le code qu'ils avaient créé pour une publication scientifique faite il y a plus de 10 ans. Malgré sa facilité apparente, ce défi est plus complexe qu'il n'y paraît. Le premier problème pourrait d'ailleurs être celui de retrouver le code source à un moment où Software Heritage n'existait pas ! Chercheur en neurosciences computationnelles cognitives à Inria et à l'institut des maladies neurodégénératives de Bordeaux, Nicolas Rougier effectue des recherches portant sur les modèles computationnels du cerveau permettant la prise de décision, l'apprentissage et la cognition. Il s'intéresse aussi aux réseaux de neurones artificiels et à l'apprentissage automatique, ainsi qu'à la recherche reproductible, la visualisation scientifique et l'informatique graphique.

L'après-midi s'est clôturée sur un exposé intitulé « Environnements logiciels reproductibles et transparents avec GNU Guix » et présenté par Ludovic Courtès (Inria). La reproductibilité des expériences impliquant du logiciel est un enjeu scientifique majeur. Pourtant celle-ci se heurte souvent à la difficulté de répliquer mais aussi d'inspecter et de modifier de manière contrôlée les environnements logiciels de ces expériences. Une réponse populaire à ce problème est l'utilisation d'outils permettant de conserver les octets qui constituent l'environnement logiciel d'une expérience. À cette approche opaque, GNU Guix oppose une approche inspirée de la programmation fonctionnelle où un déploiement logiciel à partir de son code source est vu comme un « calcul » comme un autre, et où chacun de ces calculs est une fonction pure. Ludovic a présenté ces fondements et a montré en quoi Guix est une solution utilisable aujourd'hui pour définir des environnements logiciels reproductibles bit à bit. Il a de plus abordé les implications de cette approche sur les pratiques de publication scientifique. Ingénieur de recherche Inria, Ludovic Courtès a initié le développement de GNU Guix il y a huit ans. Depuis 2017, il s'intéresse à son utilisation dans le cadre du calcul intensif (HPC) et de la science reproductible avec le projet Guix-HPC. Ludovic est un adepte de la programmation fonctionnelle, notamment en Scheme avec GNU Guile auquel il a contribué.

Cette journée a eu un franc succès avec une moyenne tournant autour de 110 téléspectateurs environ et un pic à 156 téléspectateurs.



La reproductibilité des calculs coûteux

Konrad Hinsén¹

La reproductibilité computationnelle est la possibilité de refaire un calcul en partant de ses ingrédients : code source et données d'entrée. Dans le contexte du calcul scientifique, l'intérêt de la reproductibilité est de pouvoir vérifier que des résultats d'une simulation ou d'une analyse de données, tels que publiés dans un article, proviennent réellement du calcul décrit dans ce même article, et que cette description est complète. La certitude sur la provenance des résultats permet par la suite d'explorer le code source et les données d'entrée afin de mieux comprendre ce qui a été fait.

La façon la plus évidente de vérifier la reproductibilité d'un calcul est de tenter de le refaire, et comparer à la fin les résultats avec ceux publiés par ses auteurs. Mais dans le cas d'un calcul coûteux, cette méthode n'est pas toujours applicable, parce que les moyens de calculs nécessaires sont soit indisponibles soit trop onéreux. Dans cet article, je vais décrire comment on peut se convaincre de la reproductibilité sans refaire le calcul, et aussi discuter ce que vaut une telle approche par rapport à la tentative explicite de reproduction.

C'est quoi, un calcul coûteux ?

Avant d'aller dans le vif du sujet, je vais rendre le terme abstrait du calcul coûteux plus concret. Évidemment, l'étiquette « coûteux » est subjective. Quelles sont donc les situations réelles dans lesquelles un chercheur ne peut pas simplement refaire un calcul ?

1. CR CNRS au Centre de biophysique moléculaire, Orléans.

Le cas le plus extrême est le calcul dit « haute performance », effectué en utilisant les plus grands centres de calcul actuellement disponibles : par exemple, la simulation d'une grande protéine sur le calculateur Jean Zay², installé à l'IDRIS à Orsay. Une telle simulation peut occuper des centaines de processeurs pendant quelques semaines. Et comme les logiciels sont souvent spécifiquement optimisés pour de tels ordinateurs, on ne peut pas refaire les calculs à l'identique sur une machine ordinaire, même si l'on était prêt à attendre bien plus longtemps. Reste l'option de demander un accès à ce même ordinateur (tant qu'il existe), auprès du GENCI³ qui gère les gros moyens informatique de la recherche française. Je suis sûr que les comités du GENCI n'accepteraient pas une telle demande, qui ne représente pas un bon usage de moyens de calcul rares et chers.

Dans le travail quotidien d'un chercheur, même un calcul beaucoup plus modeste peut être considéré comme coûteux. Dans le cadre d'une collaboration avec un collègue, si je souhaite régulièrement vérifier la reproductibilité de son travail afin de l'adapter par la suite à une question un peu différente, même quelques heures de calculs peuvent être un frein à la progression fluide du projet commun. En fin de compte, un calcul coûteux est un calcul qu'on ne refait pas à cause de l'effort que cela représente, même si on sait qu'on devrait le faire.

Une question de confiance

La clé pour voir comment on peut s'assurer de la reproductibilité d'un calcul sans le refaire est de reformuler l'objectif. En répétant le calcul, j'obtiens la réponse *oui* ou *non* à la question : « Moi, aujourd'hui, puis-je reproduire les résultats de ce calcul ? » Ce n'est pas vraiment la question la plus intéressante du point de vue de la science, qui est un processus collectif de longue durée. Une meilleure question serait « Sous quelles conditions, et pendant combien de temps, un chercheur compétent dans le domaine peut-il reproduire les résultats de ce calcul ? » Cette question est bien moins précise, et il est impossible d'y répondre avec une certitude absolue.

Je propose donc de remplacer cette question par une autre, plus pragmatique : « Quel événement augmenterait la confiance que l'on peut avoir dans la reproductibilité d'un calcul ? » Une répétition réussie figure sûrement sur la liste des bonnes réponses, mais il y en a d'autres, mieux adaptées aux calculs coûteux. Dans la suite de cet article, je vais me concentrer sur une réponse sur laquelle je travaille moi-même depuis une dizaine d'année : l'utilisation d'une chaîne d'outils qui garantit la reproductibilité des calculs dont on lui confie la gestion. J'appelle cette approche la *reproductibilité par construction*.

Bien évidemment, une telle chaîne d'outils doit aussi permettre une répétition explicite de tout calcul aux fins de vérification. Le but est de supprimer la *nécessité* d'une répétition, mais pas sa *possibilité*. Au contraire, la confiance dans la chaîne

2. <http://www.idris.fr/jean-zay/>.

3. <https://www.genci.fr/>.

d'outils est fondée sur la réussite sans faille de reproductions explicites pour les calculs peu coûteux.

Pourquoi la reproductibilité est si difficile ?

Pour construire une chaîne d'outils qui assure la reproductibilité, il faut d'abord comprendre quels sont les problèmes à résoudre. Le calcul étant déterministe, à l'exception de certains calculs parallèles, on s'attend à ce que la reproductibilité du calcul soit la règle plutôt que l'exception. Et si on relance le même logiciel avec les mêmes données d'entrée plusieurs fois sur le même ordinateur, on obtient en effet les mêmes résultats avec une haute fiabilité.

On peut en conclure que quand on n'arrive pas à reproduire un résultat de calcul scientifique, quelque chose a changé : le logiciel, les données d'entrée, ou l'ordinateur. Dans la vaste majorité de cas, c'est bien le logiciel qui est en cause. En fait, le « logiciel » est en réalité une pile complexe de composants, avec un système d'exploitation comme base sur laquelle on empile des couches d'infrastructure de plus en plus spécialisées, et seulement tout en haut ce que l'utilisateur considère être *son* logiciel. Au total, on a typiquement quelques centaines de composants. Il devient alors difficile d'en fournir une liste complète qui contient en plus la version précise de chaque composant. Et même avec une telle liste en mains, il n'est pas évident de reconstruire un assemblage identique plus tard et sur un autre ordinateur.

Une solution fréquemment proposée est le recours aux machines virtuelles ou aux conteneurs du genre Docker ou Singularity. Ces technologies permettent en effet de préserver une version exécutable de la totalité de la pile. Mais ils ne préservent pas le lien entre les exécutables et leur code source, qui est pourtant essentiel pour tirer un bénéfice scientifique de la reproductibilité : pouvoir inspecter le code pour le comprendre et pour l'adapter à d'autres situations.

Une autre complication s'ajoute à la complexité de la pile logicielle : tout calcul scientifique se déroule en plusieurs phases, dont chacune produit le *code* qui sera exécuté dans la phase suivante [1]. On oublie facilement les premières phases, qui mettent en oeuvre des compilateurs et d'autres outils de construction de logiciels, avant qu'on puisse lancer la dernière phase qui produit les résultats scientifiques. Mais les détails de la compilation ont souvent un impact sur ces résultats ; par exemple, via les options de compilation qui définissent les règles précises de l'arithmétique à virgule flottante. Pour rendre le calcul reproductible, il faut inclure toutes ces phases dans la reproductibilité par construction.

À la recherche de la reproductibilité par construction

Une chaîne d'outils pour assurer la reproductibilité du calcul doit donc se baser sur une fondation stable dans le temps, et permettre la construction d'une pile logicielle en plusieurs phases avec un traçage complet de chaque étape et de tout code source utilisé. Elle doit aussi être compatible avec les logiciels scientifiques en usage

aujourd'hui si elle veut avoir une utilité dans la recherche. Malheureusement, il n'est pas facile de satisfaire toutes ces exigences.

Ma première tentative [2] utilisait la *Java Virtual Machine* (JVM) comme fondation et le format HDF5⁴ pour la distribution, tant des logiciels que des données. Comme preuve de faisabilité, elle était plutôt réussie, mais peu de logiciels scientifiques sont écrits pour la JVM, et ces rares logiciels nécessitent une adaptation pour la distribution en format HDF5. Ma deuxième tentative [2] utilisait le langage Python comme base, pour pouvoir profiter de son vaste ensemble de bibliothèques scientifiques. Mais cet ensemble manque de stabilité, et les premières publications utilisant ce format en 2013 ne sont déjà plus reproductibles pour cette raison.

L'approche que j'explore actuellement est basée sur le gestionnaire de logiciels Guix⁵. Guix assure la reproductibilité d'une pile logicielle complète, mais ne prend pas en charge la dernière phase d'un calcul, celle qui produit les résultats d'intérêt scientifique. Ceci est l'objectif d'une extension en cours de développement, le *Guix Workflow Language*⁶ (GWL). En théorie, cette approche devrait satisfaire toutes les exigences. Mais eu égard à la diversité des calculs et des modes d'interaction avec l'ordinateur, un seul outil pour la dernière phase ne pourra pas satisfaire les besoins de tout le monde. Il reste du travail à faire !

Références

- [1] Konrad Hinsén, *Staged Computation : The Technique You Did Not Know You Were Using*, 2020, <http://doi.org/10.1109/MCSE.2020.2985508>.
- [2] Konrad Hinsén, *ActivePapers : a platform for publishing and archiving computer-aided research*, 2015, <http://doi.org/10.12688/f1000research.5773.3>.

4. <https://www.hdfgroup.org/solutions/hdf5/>.

5. <https://guix.gnu.org/>.

6. <https://www.guixwl.org/>.



Reproduire les environnements logiciels : un maillon incontournable de la recherche reproductible

Ludovic Courtès¹

Introduction

Un constat est largement partagé : puisque les logiciels font dorénavant partie intégrante du processus scientifique, une démarche de recherche reproductible — un pléonasme ! — doit intégrer le logiciel. Mais de quelle manière au juste ?

Le deuxième Plan national pour la science ouverte [5], publié en juillet 2021, « soutient » la diffusion du code source des logiciels de recherche sous licence libre permettant la diffusion sans restriction, mais aussi la modification et la diffusion de versions modifiées. C’est une transcription naturelle du processus scientifique : le travail de critique ne peut se faire correctement que si les pairs peuvent étudier le code source et faire leurs propres expériences. Le Plan insiste aussi sur la conservation des codes sources grâce à Software Heritage, sans quoi ce travail devient vite impossible.

Que le code source est disponible est une condition nécessaire mais pas suffisante. Les sociétés savantes ont mis en place un système de badges pour évaluer le niveau de reproductibilité des résultats décrits dans leurs publications. Celui de l’*Association for Computer Machinery* (ACM) dispose de trois niveaux selon que le code est disponible (premier niveau), est utilisable (deuxième niveau), ou que les résultats ont été

1. Ingénieur de recherche Inria, ludovic.courtes@inria.fr.

reproduits indépendamment en faisant tourner le code². La reproduction des environnements logiciels — le fait de pouvoir déployer précisément l'ensemble logiciel qui a servi à une production scientifique — est un aspect qu'on relègue volontiers au rang de détail technique mais qui est pourtant incontournable pour parvenir à cet objectif de reproductibilité. Quels outils, quelles méthodes existent pour y parvenir ?

Entre « gestion » et « gel » des environnements logiciels

Beaucoup de logiciels de recherche sont développés pour GNU/Linux et tirent parti des outils de déploiement logiciel qu'on y trouve. Les « distributions » GNU/Linux telles que Debian et Ubuntu se basent sur des outils de *gestion de paquets* comme `apt` qui permettent d'installer, de mettre à jour ou de retirer les logiciels. Ces outils ont une vision du *graphe de dépendance* des logiciels et permettent de savoir quels logiciels sont présents sur la machine.

Malheureusement, ces outils ont deux limitations : ils requièrent les droits d'administration système, et ils ne permettent de déployer qu'un seul environnement logiciel à la fois. Pour cette raison, en particulier dans le domaine du calcul intensif (*high-performance computing* ou HPC), on a développé d'autres outils de gestion de paquets destinés à être utilisés *au dessus* celui du système, avec l'avantage d'être utilisables sans les droits d'administration, par chaque utilisatrice ou utilisateur, qui peut ainsi déployer ses environnements logiciels. Les outils populaires, dans cette catégorie, incluent CONDA, Spack et EasyBuild, ainsi que des outils dédiés à un langage de programmation (pour Python, Julia, R, etc.).

L'expérience a rapidement montré que cet empilement d'outils de déploiement devient vite préjudiciable à la reproductibilité, car chaque outil ignore celui « du dessous », bien qu'il dépende de ce que celui-ci a installé. Autrement dit, chaque outil ne voit qu'une partie du graphe de dépendance de l'ensemble. À cela s'ajoute le fait que, même pris individuellement, ces outils ne permettent pas, ou difficilement, de reproduire un environnement logiciel à l'identique. C'est notamment pour cette raison qu'une deuxième approche du déploiement logiciel s'est popularisée : celle qui consiste à *geler l'environnement logiciel*, plutôt que d'essayer de le décrire.

L'idée peut se résumer ainsi : puisqu'il est difficile voire impossible de redéployer un environnement logiciel à l'identique en utilisant ces outils de gestion de paquets, créons l'environnement une fois pour toutes puis sauvegardons les octets qui le composent — les fichiers de chaque logiciel installé. On obtient ainsi une *image* binaire, qui permet, sur n'importe quelle machine et à n'importe quel moment, de relancer les logiciels scientifiques d'intérêt. Le plus souvent, on utilise les outils à base de « conteneurs » Linux, tels que Docker ou Singularity, mais une machine virtuelle peut aussi remplir cette fonction.

2. *Artifact Review and Badging – Version 1.0, Association for Computer Machinery, August 2020*

L'approche est séduisante : puisqu'on a tous les octets des logiciels, la reproductibilité est totale ; on a la garantie de pouvoir relancer les logiciels, et donc, de reproduire l'expérience scientifique. Mais l'inconvénient est de taille : puisque l'on n'a *que* les octets des logiciels, comment savoir si ces octets correspondent vraiment au code source que l'on croit exécuter ? Comment expérimenter avec cet environnement logiciel, dans le cadre d'une démarche scientifique, pour établir l'impact d'un choix de version, d'une option de compilation ou du code d'une fonction ? L'approche est pratique, c'est indéniable, mais ces deux faiblesses fragilisent l'édifice scientifique qui se bâtirait sur ces fondations.

Le déploiement logiciel vu comme une fonction pure

GNU Guix³ est un outil de déploiement logiciel qui cherche à obtenir le meilleur des deux mondes : la reproductibilité parfaite des environnements « gelés » dans des conteneurs, et la transparence et la flexibilité des outils de gestion de paquets. Il est issu de travaux à la croisée de l'ingénierie logicielle et des langages, d'abord en bâtissant sur le modèle de *déploiement purement fonctionnel* de Nix [4] et en étendant Scheme, un langage de programmation fonctionnelle de la famille Lisp, avec des abstractions permettant d'en tirer partie. En termes pratiques, Guix hérite de Nix les fondations permettant la reproductibilité d'environnements logiciels et fournit les outils pour exploiter ces capacités sans expertise préalable [3].

Le principe du déploiement fonctionnel est de traiter le processus de compilation d'un logiciel comme une *fonction pure*, au sens mathématique : les entrées de la fonction sont le code source, un compilateur et des bibliothèques, et son résultat est le logiciel compilé. Les mêmes entrées mènent au même résultat ; Guix s'assure que c'est effectivement le cas en vérifiant que les compilations sont reproductibles *au bit près*, ainsi que le font d'autres projets participant à l'effort *Reproducible Builds*⁴. Cette approche fonctionnelle capture ainsi l'essence de la variabilité du déploiement logiciel. Car, en fait, une série de noms et de numéros de version de logiciels *ne suffit pas* à rendre compte des variations qui peuvent être introduites lors du déploiement de chaque logiciel.

Chaque déploiement logiciel est vu comme une fonction pure, la définition est donc récursive. Mais puisqu'elle est récursive, qu'y a-t-il « au tout début » ? Quel compilateur compile le premier compilateur ? On arrive là à une question fondamentale, presque philosophique, mais qui a un impact très concret sur la transparence des systèmes logiciels comme l'a expliqué Ken Thompson dans son allocution pour la remise du prix Alan Turing [7] : tant que subsistent dans le graphe de dépendance des binaires opaques dont la provenance ne peut pas être vérifiée, il est impossible d'établir avec certitude l'authenticité des binaires qui en découlent.

3. GNU Guix, <https://guix.gnu.org>.

4. *Reproducible Builds*, <https://reproducible-builds.org>.

Guix s'attaque à ce problème en basant le graphe de dépendance de ses paquets sur un ensemble de binaires pré-compilés bien identifié et le plus petit possible — actuellement quelques dizaines de méga-octets —, avec pour objectif de le réduire à un seul binaire suffisamment petit pour pouvoir être analysé par un humain [6]. C'est là un sujet d'ingénierie et de recherche à part entière.

Déclarer et reproduire un environnement logiciel

Guix peut s'utiliser comme une distribution à part entière avec Guix System ou alors comme un outil de déploiement par dessus une distribution existante et donnant accès à plus de 18 000 logiciels libres. Il fournit une interface en ligne de commande similaire à celle des outils de gestion de paquets : la commande `guix install python`, par exemple, installe l'interprète Python, la commande `guix pull` met à jour la liste des logiciels disponibles et `guix upgrade` met à jour les logiciels précédemment installés. Chaque opération s'effectue sans les droits d'administration système. Plusieurs supercalculateurs en France et à l'étranger proposent Guix et le projet Guix-HPC, qui implique plusieurs institutions dont Inria, vise à élargir le mouvement⁵.

Voyons maintenant comment on peut concrètement, en tant que scientifique, utiliser cet outil pour que ses expériences calculatoires soient reproductibles. On peut commencer par lister dans un *manifeste* les logiciels à déployer ; ce manifeste peut être partagé avec ses pairs et stocké en gestion de version. L'exemple ci-dessous nous montre un manifeste pour les logiciels Python, SciPy et NumPy :

```
(specifications->manifest
' ("python" "python-scipy" "python-numpy"))
```

Il s'agit de code Scheme. Une utilisation avancée serait par exemple d'inclure dans le manifeste des définitions de paquets ou de variantes de paquets ; Guix permet notamment de réécrire facilement le graphe de dépendance d'un paquet pour le personnaliser, ce qui est une pratique courante en HPC [3].

Supposons que l'on ait ainsi créé le fichier `manifeste.scm`, on peut déployer les logiciels qui y sont listés — et uniquement ceux-là — avec, par exemple, la commande suivante :

```
guix package -m manifeste.scm
```

Ce manifeste, toutefois, ne contient que des noms de paquets symboliques, et pas de numéros de version, options de compilation, etc. Comment dans ces conditions reproduire exactement le même environnement logiciel ?

Pour cela, il nous faut une information supplémentaire : l'identifiant de révision de Guix. Puisque Guix et toutes les définitions de paquets qu'il fournit sont stockées dans un dépôt de gestion de version Git, l'identifiant de révision désigne de manière

5. Projet Guix-HPC, <https://hpc.guix.info>.

non ambiguë *l'ensemble du graphe de dépendance des logiciels* — aussi bien la version de Python, que ses options des compilations, ses dépendances, et ceci récursivement jusqu'au compilateur du compilateur. C'est la commande `guix describe` qui donne la révision actuellement utilisée :

```
$ guix describe
Génération 107 20 juil. 2021 13:23:35 (actuelle)
guix 7b9c441
URL du dépôt : https://git.savannah.gnu.org/git/guix.git
branche : master
commit : 7b9c4417d54009efd9140860ce07dec97120676f
```

En conservant cette information adossée au manifeste, on a de quoi reproduire *au bit près* cet environnement logiciel, sur des machines différentes, mais aussi à des moments différents. Le plus pratique est de stocker cette information dans un fichier, au format que Guix utilise pour représenter les *canaux* utilisés⁶ :

```
guix describe -f channels > canaux.scm
```

Une personne souhaitant reproduire l'environnement logiciel pourra le faire avec la commande suivante :

```
guix time-machine -C canaux.scm -- package -m manifeste.scm
```

Cette commande va d'abord obtenir et déployer la révision de Guix spécifiée dans `canaux.scm`, à la manière d'une « machine à voyager dans le temps ». C'est ensuite la commande `guix package` de cette révision là qui est lancée pour déployer les logiciels conformément à `manifeste.scm`. En quoi est-ce différent d'autres outils ?

D'abord, une version ultérieure de Guix peut reproduire une ancienne version de Guix et de là, déployer les logiciels décrits dans le manifeste. Contrairement à des outils comme CONDA ou `apt`, Guix ne repose pas sur la mise à disposition de binaires pré-compilés sur les serveurs du projet ; il peut utiliser des binaires pré-compilés — et ça rend les installations plus rapides — mais ses définitions de paquets contiennent toutes les instructions nécessaires pour compiler chaque logiciel, avec un résultat déterministe au bit près.

Une des deux différences majeures par rapport à l'approche qui consiste à geler un environnement dans une image Docker ou similaire est le *suivi de provenance* : au lieu de binaires inertes, on a là accès au graphe de dépendance complet lié au code source des logiciels. Chacune ou chacun peut *vérifier* que le binaire correspond bien au code source — puisque les compilations sont déterministes — plutôt que de faire confiance à la personne qui fournit les binaires. C'est le principe même de la démarche scientifique expérimentale qui est appliquée au logiciel.

La deuxième différence est que, ayant accès à toutes les instructions pour compiler les logiciels, Guix fournit aux usagers les moyens d'*expérimenter* avec cet

6. Un *canal* Guix est une collection de définitions de paquets stockée dans un dépôt Git.

ensemble logiciel : on peut, y compris depuis la ligne de commande, faire varier certains aspects, tel que les versions ou variantes utilisées. L'expérimentation reste possible.

Et si le code source de ces logiciels venait à disparaître ? On peut compter sur Software Heritage (SWH en abrégé), qui a pour mission rien de moins que d'archiver tout le code source public disponible⁷. Depuis quelques années, Guix est intégré à SWH de deux manières : d'une part Guix va automatiquement chercher le code source sur SWH lorsqu'il est devenu indisponible à l'adresse initiale [1], et d'autre part Guix alimente la liste des codes sources archivés par SWH. Le lien avec l'archivage de code en amont est assuré.

Vers des articles reproductibles

On a vu le lien en amont avec l'archivage de code source, mais le lien en aval avec la production scientifique est également crucial. C'est un travail en cours, mais on peut déjà citer quelques initiatives pour construire au-dessus de Guix des outils et méthodes pour aider les scientifiques dans la production et dans la critique scientifique.

Alors que les annexes pour la reproductibilité logicielle des conférences scientifiques sont bien souvent informelles, écrites en langage naturel, et laissent le soin aux lecteurs et lectrices de reproduire tant bien que mal l'environnement logiciel, un saut qualitatif consiste à fournir les fichiers `canaux.scm` et `manifeste.scm` qui constituent en quelque sorte une description exécutable de l'environnement logiciel. Cette approche a montré ses bénéfices notamment pour des travaux en génomique dont les résultats demandent de nombreux traitements logiciels.

La revue en ligne ReScience C organisait en 2020 le *Ten Years Reproducibility Challenge*, un défi invitant les scientifiques à reproduire les résultats d'articles vieux de dix ans ou plus⁸. C'est dans ce cadre que nous avons montré comment Guix peut être utilisé pour décrire l'ensemble d'une chaîne de traitement scientifique, incluant le code source du logiciel dont traite l'article, les expériences effectuées avec ce logiciel, les courbes produites dans ce cadre, pour enfin arriver au PDF de l'article incluant la prose et ces courbes [2]. Toute cette chaîne est décrite, automatisée, et reproductible, de bout en bout. Cette approche pourrait être généralisée aux domaines scientifiques ne requérant pas de ressources de calcul spécialisées ; nous comptons fournir des outils pour la rendre plus accessible.

La question du déploiement logiciel se retrouve également dans d'autres contextes. Guix-Jupyter⁹, par exemple, permet d'ajouter à des bloc-notes Jupyter des annotations décrivant l'environnement logiciel dans lequel doit s'exécuter le

7. Software Heritage, <https://www.softwareheritage.org>.

8. ReScience, *Ten Years Reproducibility Challenge*, <https://rescience.github.io/ten-years/>.

9. Guix-Jupyter, <https://gitlab.inria.fr/guix-hpc/guix-kernel>

bloc-notes. L'environnement décrit est automatiquement déployé *via* Guix, ce qui garantit que les cellules du bloc-notes s'exécutent avec les « bons » logiciels. Dans le domaine du calcul intensif et du traitement de données volumineuses, le *Guix Workflow Language* (GWL) permet de décrire des chaînes de traitement pouvant s'exécuter sur des grappes de calcul tout en bénéficiant de déploiement reproductible *via* Guix¹⁰.

Adapter les pratiques scientifiques

La place croissante prise par le logiciel dans les travaux scientifiques, paradoxalement, avait probablement été une des causes de la « crise » de la reproductibilité en sciences expérimentales que beaucoup ont observée — par la perte de bonnes pratiques anciennes telles que les cahiers de laboratoire. Notre souhait est qu'elle puisse maintenant, au contraire, permettre une *meilleure* reproductibilité des résultats expérimentaux, en maintenant à tout prix la rigueur scientifique quand on arrive dans le terrain logiciel.

De même que les outils de gestion de version sont progressivement entrés dans la boîte à outils des scientifiques comme un moyen incontournable de suivre les évolutions d'un logiciel, les outils de déploiement logiciel reproductible tels que Guix mériteraient faire partie des bonnes pratiques communément admises. Il en va de la crédibilité de la démarche scientifique moderne.

Références

- [1] Ludovic Courtès. Connecting reproducible deployment to a long-term source code archive, March 2019. <https://guix.gnu.org/en/blog/2019/connecting-reproducible-deployment-to-a-long-term-source-code-archive/>.
- [2] Ludovic Courtès. [Re] Storage tradeoffs in a collaborative backup service for mobile devices. *ReScience C*, 6(1), June 2020. <https://doi.org/10.5281/zenodo.3886739>.
- [3] Ludovic Courtès and Ricardo Wurmus. Reproducible and user-controlled software environments in HPC with Guix. In *Euro-Par 2015 : Parallel Processing Workshops*, Lecture Notes in Computer Science, pages 579–591, August 2015.
- [4] Eelco Dolstra, Merijn de Jonge, and Eelco Visser. Nix : A safe and policy-free system for software deployment. In *Proceedings of the 18th Large Installation System Administration Conference (LISA '04)*, pages 79–92. USENIX, November 2004.
- [5] Ministère de l'enseignement supérieur, de la recherche et de l'innovation. Deuxième Plan national pour la science ouverte — Généraliser la science ouverte en France, 2021–2024, July 2021. <https://www.ouvrirelascience.fr/deuxieme-plan-national-pour-la-science-ouverte/>.
- [6] Jan Nieuwenhuizen. Guix further reduces bootstrap seed to 25%, June 2020. <https://guix.gnu.org/en/blog/2020/guix-further-reduces-bootstrap-seed-to-25/>.
- [7] Ken Thompson. Reflections on trusting trust. *Communications of the ACM*, pages 761–763, August 1984.

10. Guix Workflow Language, <https://workflows.guix.info>.



L'ordinateur quantique : un objectif impossible à atteindre ?

Michel Dyakonov¹

Cet article a déjà fait l'objet d'une publication sur le blog de Mediapart² et est une réaction personnelle à la stratégie nationale sur les technologies quantiques présentée par le Président de la République à l'université Paris-Saclay³ le 21 janvier 2021.

Depuis près de 30 ans, nous assistons à une ruée mondiale vers « l'informatique quantique » accompagnée d'un battage médiatique d'un niveau sans précédent. Les fonds sont généreusement distribués, des centres d'informatique quantique s'ouvrent partout dans le monde et des perspectives à couper le souffle sont présentées au profane par des scientifiques enthousiastes et des journalistes encore plus enthousiastes. L'impression a été créée que l'informatique quantique sera la prochaine percée technologique du XXI^e siècle. Le grand public, ainsi que les agences de financement, sont convaincus que la révolution quantique, apportant la finance quantique, l'internet quantique, la sécurité quantique, la santé quantique et d'autres miracles quantiques, est à notre porte.

Le mot « quantum » est entré dans le vocabulaire des foyers et des rock stars et a acquis le sens de merveilleux, moderne, puissant, cool et génial. Il est maintenant librement utilisé par les journalistes, les politiciens et toutes sortes de représentants du gouvernement qui n'ont absolument aucune idée du vrai sens de ce mot, ni de la

1. Physicien, professeur émérite à l'université de Montpellier, membre d'honneur de l'institut physico-technique à Saint-Pétersbourg.

2. <https://blogs.mediapart.fr/edition/au-coeur-de-la-recherche/article/220221/lordinateur-quantique-un-objectif-impossible-atteindre>.

3. <http://www.c2n.universite-paris-saclay.fr/fr/visite-m-macron>.

physique quantique en général. La science de « l'informatique quantique et le calcul quantique » est enseignée dans de nombreuses universités et écoles d'ingénieurs à travers le monde (dont la France) attirant les meilleurs étudiants qui rêvent de contribuer à cette entreprise audacieuse⁴.

C'est devenu une course aux armements qui s'auto-entretient, de nombreuses organisations et institutions restant apparemment dans la course, ne serait-ce que pour éviter d'être laissées pour compte. Certains des plus grands scientifiques du monde, chez Google, IBM, Intel, Microsoft et d'autres, travaillent dur et avec des ressources somptueuses dans des laboratoires à la pointe de la technologie, pour réaliser leur vision d'un avenir quantique.

Bref historique

L'algorithme de Shor. Les premières idées assez vagues de l'informatique quantique ont été exprimées par Paul Benioff (1980), Yuri Manin (1980) et Richard Feynman (1981), mais le sujet n'a vraiment attiré l'attention qu'en 1994 lorsque Peter Shor [9] a proposé un algorithme qui, avec un ordinateur quantique idéal, pourrait factoriser des nombres extrêmement grands beaucoup plus rapidement que l'ordinateur classique. Ce résultat mathématique exceptionnel a engendré tout le domaine de l'informatique quantique ! L'excitation a été causée par la possibilité pour le futur ordinateur quantique de casser des codes de sécurité, dont certains, mais pas tous, sont basés sur l'énorme difficulté, voire l'impossibilité, pour les ordinateurs conventionnels de factoriser des nombres à 1000 chiffres, qui sont des produits de très grands nombres premiers⁵. Plusieurs autres algorithmes quantiques ont également été proposés plus tard.

Factorisation expérimentale de 15 par l'algorithme de Shor. La première expérience consacrée à la factorisation de 15 par Shor a été rapportée par Vandersypen et al. [12] en utilisant la technique de résonance magnétique nucléaire (RMN). Toutes les portes ont été mises en œuvre par des impulsions micro-ondes appliquées en 1 seconde environ, ce qui est inférieur au temps de décohérence nucléaire. Les spectres RMN obtenus correspondaient très bien aux prédictions de la procédure de Shor.

Lanyon et al. [4] ont effectué la même tâche dans une expérience optique utilisant la polarisation de quatre photons, tandis que Lucero et al. [5] ont utilisé des qubits. Selon Josephson, « ... nous exécutons une version compilée de trois qubits de l'algorithme de Shor pour factoriser le nombre 15, et trouver avec succès les facteurs premiers 48 % du temps ».

4. Les âmes innocentes ne savent pas encore que dans un avenir pas très lointain, il pourrait devenir prudent de dissimuler leur éducation en « informatique quantique ».

5. Notez que 27 ans plus tard, la démonstration de $15 = 3 \times 5$ par l'algorithme de Shor n'est toujours pas possible (voir plus loin). Nous n'avons donc pas de raison de nous inquiéter pour nos codes de sécurité.

Mais, dans toutes ces expériences, la soi-disant version compilée de l'algorithme de Shor a été utilisée. Comme l'ont montré Beckman et al. [1], l'algorithme complet peut factoriser un nombre de k bits en utilisant de l'ordre de $72k^3$ portes quantiques élémentaires. Par exemple, la factorisation de 15 nécessite 4608 portes fonctionnant sur 21 qubits. Reconnaisant que ces exigences vont bien au-delà des possibilités expérimentales, Beckman et al. ont introduit une technique de compilation qui exploite les propriétés du nombre à factoriser, permettant l'exploration de l'algorithme de Shor avec un nombre considérablement réduit de ressources.

On pourrait dire que c'est une sorte de triche (innocente) sachant à l'avance que $15 = 3 \times 5$, on peut prendre quelques raccourcis, ce qui ne serait pas possible si le résultat n'était pas connu à l'avance.

Tous les tests expérimentaux existants de l'algorithme de Shor utilisent cette approche simplifiée. Dans un travail très remarquable de Martin-Lopez et al. [7], la même approche a permis pour la première fois de factoriser 21 dans une expérience optique, où une procédure itérative de recyclage d'un seul qubit a été mise en œuvre avec succès.

La simple factorisation du nombre 15 en utilisant l'algorithme de Shor complet est encore bien au-delà de la portée des possibilités expérimentales (comme je l'ai prédit en 2001 [2]). Ainsi, il semble que dans un avenir prévisible, nous n'avons pas à nous soucier de la sécurité des codes de cryptographie en raison de la difficulté à factoriser de très grands nombres, qui sont des produits de nombres premiers à cent chiffres.

Cartes routières

Le document récent « stratégie nationale sur les technologies quantiques » est en fait une feuille de route détaillée sur cinq ans pour la recherche sur l'informatique quantique et ses applications avec l'objectif annoncé : « 5 ans pour asseoir la France dans le premier cercle mondial ».

Auparavant, des feuilles de route « quantiques » similaires avaient déjà été créées, mais n'ont jamais été réalisées concrètement. La première est apparue en 2002 lorsque, à la demande de l'agence *Advanced Research and Development Activity* (ARDA) du gouvernement américain, une équipe d'éminents experts de l'information quantique avait établi une feuille de route audacieuse [12] pour l'informatique quantique avec pour objectifs à cinq ans (d'ici 2007) :

- coder un seul qubit dans l'état d'un qubit logique formé de plusieurs qubits physiques ;
- effectuer une correction d'erreur répétitive du qubit logique ;
- transférer l'état du qubit logique dans l'état d'un autre ensemble de qubits physiques avec une grande fidélité ;

et un objectif à 10 ans (d'ici 2012) : implémenter un code correcteur d'erreurs quantiques concaténé⁶.

Alors qu'un jury très bienveillant pourrait considérer que le premier des objectifs de 2007 est en partie atteint à présent, les attentes pour les autres objectifs de 2007, et pour l'objectif de 2012, sont reportées *sine die*. Il en va de même pour les autres prédictions : « Au fur et à mesure que les ordinateurs quantiques à plus grande échelle seront développés au cours des cinq et dix prochaines années, la simulation quantique continuera probablement à être l'application pour laquelle les ordinateurs quantiques peuvent apporter des améliorations substantielles par rapport au calcul classique ».

Cependant, rien de ressemblant, même de loin, à des ordinateurs quantiques à plus grande échelle, ni aucune amélioration par rapport au calcul classique, n'a été produit jusqu'à présent.

Apparemment, passer de 5 qubits à 50 (l'objectif fixé par la feuille de route du panel d'experts ARDA pour l'année 2012) présente des difficultés expérimentales pratiquement insurmontables. Ils sont très probablement liés au simple fait que $2^5 = 32$, tandis que $2^{50} = 1125899906842624$ (voir ci-dessous l'explication de l'augmentation exponentielle de la difficulté à augmenter le nombre de qubits).

Quatorze ans plus tard (mai 2016), un groupe de scientifiques européens renommés a contacté la Commission européenne avec un « Manifeste quantique » [4], un appel à lancer une initiative européenne ambitieuse dans les technologies quantiques, nécessaire pour assurer le rôle de premier plan de l'Europe dans la deuxième révolution quantique en cours dans le monde entier. La Commission semble avoir été assez sensible à cet appel⁷ et a alloué 1 milliard d'euros⁸ pour atteindre cet objectif. Les résultats attendus pour l'informatique quantique sont :

- 0 – 5 ans : fonctionnement d'un qubit logique protégé par correction d'erreur ou topologiquement ; nouveaux algorithmes pour ordinateur quantique (4 ans se sont déjà écoulés, sans résultats visibles !);
- 5 à 10 ans : petit processeur quantique exécutant des algorithmes technologiquement pertinents ; résolution de problèmes de chimie et de science des matériaux avec un ordinateur quantique à usage spécial ;
- supérieur à 10 ans : intégration du circuit quantique et du matériel de contrôle classique cryogénique ; les ordinateurs quantiques à usage général dépassent la puissance de calcul des ordinateurs classiques.

On peut remarquer une similitude entre les projections à 5 et 10 ans des feuilles de route 2002 et 2016, d'autant plus que l'objectif quinquennal 2016 du Manifeste

6. La « concaténation » dans ce contexte signifiait distribuer l'information d'un qubit dans l'état d'un certain ensemble de plusieurs qubits, de sorte que le qubit logique soit codé par plusieurs qubits physiques.

7. Les américains dépensent des milliards de dollars sur ce projet, ils ne feraient rien de complètement stupides n'est-ce pas ?

8. Il apparaît que 1 milliard est l'unité quantique élémentaire pour le financement de la recherche en information quantique.

quantique est en fait le même que l'objectif quinquennal ARDA 2002. Verrons-nous une autre feuille de route similaire en 2030, avec un objectif de 5 ans d'encodage de qubits simples dans l'état d'un qubit logique ?

Plus récemment, fin 2018, un autre groupe d'experts réuni par les *National Academies of Science, Engineering and Medicine* des États-Unis a publié un rapport détaillé de 205 pages discutant de certains des défis pour lesquels l'informatique quantique serait une technologie de valeur [5]. Les auteurs du rapport déclarent sèchement qu'au cours de la prochaine décennie, aucun ordinateur quantique ne sera capable de casser des codes cryptographiques basés sur la factorisation des nombres premiers (un exemple d'une tâche pour laquelle les ordinateurs quantiques sont censés être particulièrement bien adaptés⁹), ils ne donnent aucun avis sur la possibilité de casser ces chiffres dans un avenir plus lointain.

Pour l'instant, il n'existe toujours aucun dispositif quantique capable de faire de l'arithmétique élémentaire comparable aux capacités d'un abaque, et encore moins de surpasser la règle à calcul ou la plus simple calculatrice électronique, dans tout type de calcul.

Stratégie nationale sur les technologies quantiques (2021)

Il semble très probable que ce document suivra les feuilles de route précédentes citées ci-dessus, qui, n'ayant aucun rapport avec la réalité, sont toutes allées à la poubelle. Regardons de plus près ce nouveau projet. Cela commence par la déclaration péremptoire suivante [6] : « *Le raccourcissement des temps de calcul d'un facteur de un milliard, qu'apporteront les ordinateurs quantiques d'ici 5 à 10 ans, constitue une rupture technologique majeure* ».

Le facteur d'un milliard, c'est vraiment beaucoup ! Un raccourcissement des temps de calcul d'un facteur de 2 seulement serait déjà assez important, mais pour le moment, il n'y a absolument aucune raison, même pour des attentes aussi modestes, et encore moins le facteur d'un milliard. Le projet se poursuit avec une liste d'impacts socio-économiques :

- mieux se soigner ;
- mieux se nourrir ;
- mieux combattre le changement climatique et ses effets ;
- mieux anticiper les catastrophes naturelles ;
- mieux se déplacer ;
- mieux produire ;
- mieux se protéger des menaces sur la sécurité des communications ;
- mieux se préparer aux conflits de demain.

9. Nous rappelons qu'il y a 27 ans, cette idée a été le déclencheur de la recherche en informatique quantique.

Comme nous le savons, il n'est toujours pas possible d'utiliser le calcul quantique pour vérifier que $15 = 3 \times 5$. Au vu de cela, on a du mal à croire que dans les cinq prochaines années, les ordinateurs quantiques pourraient vraiment nous aider à « combattre le changement climatique et ses effets », voire à « mieux se nourrir »...

Puisqu'aucune responsabilité n'est engagée, pour augmenter encore l'effet comique, on pourrait facilement ajouter de nombreux autres « impacts socio-économiques » tels que :

- mieux passer le bac en candidat libre ;
- mieux se marier ;
- mieux élever les enfants ;
- etc.

Mesdames, messieurs, auteurs de ce projet, un peu d'humain s'il vous plaît !

La section « Les objectifs clés » définit les principaux objectifs à fixer d'ici cinq ans : « *maîtriser les technologies quantiques offrant un avantage stratégique décisif, dont les accélérateurs, simulateurs et ordinateurs quantiques, les logiciels métiers pour le calcul quantique, les capteurs, les systèmes de communication.* »

Dans le domaine du calcul, le thème central de la stratégie : devenir le premier Etat à disposer d'un prototype complet d'ordinateur quantique généraliste de première génération dès 2023 [...]. »

La question est de savoir si le prototype complet d'ordinateur quantique généraliste de première génération sera capable de prouver que $15 = 3 \times 5$, ou en général de faire quelque chose de raisonnable, aussi simple cela soit-il. Sinon (comme le suggère notre expérience des précédentes feuilles de route « quantiques »), il ne méritera guère le titre de « prototype complet d'ordinateur quantique généraliste ».

Le document décrit également « Les faits marquants dans l'écosystème quantique national, ces 6 derniers mois ». Voici une liste de différents résultats dont certains sont certes impressionnants, mais pas directement liés à la possibilité de construire un ordinateur quantique. Parmi eux, le simulateur quantique amélioré *Quantum Learning Machine* d'Atos (QLM), qui est en fait un puissant ordinateur classique qui sert à simuler un petit ordinateur quantique, ce qui pourrait être assez intéressant.

Ordinateurs NISQ

La part du lion du financement total (352 M€) est censée être consacrée aux ordinateurs *Noisy Intermediate Scale Quantum* (NISQ, abréviation introduite par John Preskill). Pour comprendre ce que cela signifie, remplacez Q par C : que pourrait bien vouloir dire *ordinateur classique bruyant à l'échelle intermédiaire* ? De toute évidence, ce serait juste un assemblage qui ne peut rien calculer du tout et dont la destination naturelle serait la poubelle ; il en va de même pour un ordinateur NISQ.

Une expression plus polie du même jugement sur les ordinateurs NISQ a été faite dans le rapport du groupe d'experts réuni par les *National Academies of Science, Engineering and Medicine* des États-Unis (2018), mentionné ci-dessus : « Il n'existe actuellement aucun algorithme connu ni application qui pourrait utiliser efficacement cette classe de machine ».

Qubits vs bits : le principal problème de l'informatique quantique

Nos ordinateurs classiques sont essentiellement un assemblage d'interrupteurs marche / arrêt très rapides, réalisés physiquement initialement avec des tubes à vide, et plus tard, en utilisant de minuscules transistors. Les performances de l'ordinateur consistent à faire fonctionner ces interrupteurs selon un programme donné. Chaque commutateur représente un bit d'information oui / non.

À un moment donné, l'état de l'ordinateur classique est décrit par une séquence ($\uparrow\downarrow\uparrow\uparrow\downarrow\dots$), où \uparrow et \downarrow représentent des bits d'information matériellement réalisés comme les états marche et arrêt de transistors individuels. Avec n transistors, il y a 2^n états différents possibles de l'ordinateur, ce qui devient rapidement gigantesque quand n devient grand. Le processus de calcul consiste en une séquence de commutation de certains transistors entre leurs états \uparrow et \downarrow selon un programme prescrit.

En informatique quantique, l'élément classique à deux états est remplacé par un élément quantique à deux états de base, que nous représentons à nouveau symboliquement par \uparrow et \downarrow . Dans la littérature, un objet quantique arbitraire avec deux états de base est appelé un « qubit » (i.e. bit quantique).

Le choix des états de base \uparrow et \downarrow est une question de convention, et ces états ne sont pas les seuls possibles. L'état général du qubit est décrit par la fonction d'onde :

$$(1) \quad \psi = a \uparrow + b \downarrow$$

où les amplitudes quantiques a et b sont des nombres complexes arbitraires satisfaisant la condition de normalisation

$$(2) \quad |a|^2 + |b|^2 = 1$$

De plus, $|a|^2$ et $|b|^2$ donnent les probabilités que le qubit soit dans les états \uparrow et \downarrow , respectivement.

Il est important de comprendre que les valeurs de a et b dépendent de notre choix des états de base et qu'il existe un nombre infini de possibilités équivalentes pour le faire (similaire au nombre infini de possibilités de choisir les axes x et y dans le plan).

Contrairement au bit classique qui ne peut être que dans l'un des deux états marche ou arrêt, \uparrow ou \downarrow , le qubit peut être dans un continuum d'états définis par

les amplitudes quantiques a et b . Contrairement au bit classique, qui est un interrupteur marche / arrêt, le qubit est un objet continu ! Deux qubits ont quatre états de base : $\uparrow\uparrow$, $\uparrow\downarrow$, $\downarrow\uparrow$ et $\downarrow\downarrow$; ainsi l'état général de deux qubits est décrit par la fonction d'onde :

$$(3) \quad \square\square = a \uparrow\uparrow + b \uparrow\downarrow + c \downarrow\uparrow + d \downarrow\downarrow$$

avec quatre amplitudes quantiques a , b , c , et d , contraintes par la condition de normalisation $|a|^2 + |b|^2 + |c|^2 + |d|^2 = 1$ (et, bien sûr $|a|^2$ est la probabilité d'être dans l'état $\uparrow\uparrow$, idem pour les autres états).

Avec n qubits, il y a 2^n états de base, donc l'état général de n qubits est défini par une fonction d'onde avec 2^n amplitudes quantiques, qui sont des nombres complexes arbitraires restreints par la condition de normalisation uniquement. En revanche, un système classique de n pointeurs qui peuvent tourner librement autour de points fixes, est défini par 2^n paramètres (deux angles polaires pour chaque pointeur). Par exemple, 10 pointeurs sont décrits par $2 \times 10 = 20$ paramètres continus, tandis que 10 qubits sont caractérisés par $2^{10} = 1024$ paramètres continus. Évidemment, il est considérablement plus difficile de garder sous contrôle 1024 paramètres que 20 paramètres !

Ainsi, l'ordinateur quantique hypothétique est une machine analogique. Il a un nombre infini (continuum) d'états, définis par les valeurs de 2^n amplitudes complexes qui sont des variables continues. C'est la raison principale pour laquelle le calcul quantique est impossible : pour un nombre raisonnable de qubits ($n = 100$ ou 1000) le nombre de paramètres continus définissant l'état de notre machine (2^n) est incroyablement énorme, et en présence de bruit, erreurs, etc., il ne sera jamais possible de les garder toutes sous notre contrôle. On peut dire qu'il est exponentiellement plus difficile de construire un ordinateur quantique qu'un ordinateur analogique classique.

Correction des erreurs

En réponse à des préoccupations similaires, Peter Shor [10] et Andrew Steane [11] ont proposé une méthode de correction d'erreur quantique, basée sur la redondance, qui est une généralisation de la méthode de correction d'erreur dans les ordinateurs conventionnels (classiques), mais beaucoup plus sophistiquée. Il est largement reconnu que le calcul quantique sans correction d'erreur efficace est impossible. Expérimentalement, il existe plusieurs articles décrivant la « preuve de principe » de correction d'erreur d'un seul qubit. Le résultat le plus récent, plutôt modeste [8] consiste à augmenter la durée de vie d'un qubit corrigé unique à 320 microsecondes, soit environ deux fois plus que la durée de vie d'un qubit non corrigé. Jusqu'à présent, 25 ans après l'avancée du concept de correction d'erreur quantique, il n'existe

toujours pas de dispositif quantique, qui effectue une correction d'erreur efficace, même à très petite échelle.

Conclusion

Le champ gigantesque de l'informatique quantique avec plusieurs milliers de chercheurs actifs, des centaines de milliers de publications, de nombreuses conférences, écoles et ateliers, des annonces quotidiennes de nouvelles percées et de nombreux milliards de dollars dépensés, a été déclenché par l'invention de Shor de son célèbre algorithme de factorisation de très grands nombres (ouvrant ainsi éventuellement la porte aux ordinateurs quantiques pour casser les codes de sécurité), et par le développement de méthodes de correction d'erreur quantique, généralement considérées comme absolument indispensables.

L'euphorie mondiale de l'informatique quantique et l'excitation générale durent déjà depuis 27 ans ! Avant de s'engager pendant encore un quart de siècle, il serait peut-être judicieux de jeter un œil aux réalisations atteintes à ce jour au cours de cette période.

Le résultat observable peut se résumer comme suit :

- la factorisation du nombre 15 par l'algorithme de Shor n'est toujours pas possible ;
- la correction d'erreur n'a toujours pas été réalisée, même à très petite échelle (le calcul quantique sans correction d'erreur efficace est généralement considéré comme impossible) ;
- aucun dispositif quantique n'existe, capable de faire de l'arithmétique élémentaire, comme 3×5 ou $3 + 5$.

Ainsi, après un quart de siècle, il n'y a pas de résultats significatifs en informatique quantique. Les seules machines quantiques fonctionnelles à ce jour sont celles introduites par la société D-wave Systems en 1999, et actuellement étudiées et développées de manière intensive par Amazon, Google, IBM et d'autres géants de la technologie, ainsi que par la société D-wave elle-même. Ces machines peuvent effectuer un « recuit quantique » mais jusqu'à présent ne sont pas capables de correction d'erreurs et ne sont donc *pas* des ordinateurs quantiques au sens généralement accepté de ce terme. Elles non plus ne sont pas capables de factoriser 15, ni de calculer $3 + 5$. Cependant, ces machines sont intéressantes du point de vue scientifique et permettent d'obtenir des résultats intéressants.

En l'absence de raisons claires de croire que cette situation va changer au cours des prochaines décennies, les perspectives de l'informatique quantique semblent extrêmement douteuses.

Le scepticisme est une attitude normale et saine en science, par opposition à la religion, et il appartient au croyant de donner une preuve convaincante que le miracle

attendu est sur le point de se produire. Après 27 ans d'efforts considérables d'une armée de chercheurs, une telle preuve est toujours absente.

Le niveau sans précédent de battage médiatique et de promesses infondées accompagnant l'entreprise de calcul quantique n'est pas non plus un bon signe, de même que la multitude de propositions pour la plupart assez irresponsables du type « calcul quantique avec... ». Il est vraiment dommage qu'ils ne contiennent jamais, comme il se doit, le fameux avertissement de Landauer [2] : « Ce schéma, comme tous les autres schémas de calcul quantique, repose sur une technologie spéculative, ne prend pas en compte dans sa forme actuelle toutes les sources possibles de bruit, de manque de fiabilité et d'erreur de fabrication, et ne fonctionnera probablement pas ».

En faisant du vélo, après quelques entraînements, nous apprenons à contrôler avec succès trois degrés de liberté : la vitesse, la direction et l'angle que notre corps fait par rapport au trottoir. Un artiste de cirque parvient à conduire un vélo à une roue avec quatre degrés de liberté. Maintenant, imaginez un vélo ayant 1000 (ou plutôt 2^{1000}) articulations qui permettent des rotations libres de leurs pièces les unes par rapport aux autres. Quelqu'un sera-t-il capable de conduire cette machine ?

Il semble assez évident que nous n'aurons jamais d'ordinateur quantique.

Au lieu de cela, nous pourrions avoir des dispositifs quantiques spéciaux extrêmement coûteux fonctionnant à des températures de l'ordre du millikelvin. La saga de l'informatique quantique attend une analyse sociologique approfondie, et quelques leçons pour l'avenir devraient être tirées de cette aventure fascinante.

Une discussion plus détaillée du projet d'informatique quantique peut être trouvée dans mon livre [3].

Références

- [1] David Beckman, Amalavoyal N. Chari, Srikrishna Devabhaktuni, and John Preskill. Efficient networks for quantum factoring. *Phys. Rev. A*, 54 :1034–1063, Aug 1996.
- [2] M I Dyakonov. Quantum computing : A view from the enemy camp. *Optics and Spectroscopy*, 95(2) :261–267, 2003.
- [3] Mikhail I. Dyakonov. *Will We Ever Have a Quantum Computer ?* Springer, Cham, 2020.
- [4] B. P. Lanyon, T. J. Weinhold, N. K. Langford, M. Barbieri, D. F. V. James, A. Gilchrist, and A. G. White. Experimental demonstration of a compiled version of shor's algorithm with quantum entanglement. *Phys. Rev. Lett.*, 99 :250505, Dec 2007.
- [5] Erik Lucero, R Barends, Y Chen, J Kelly, M Mariantoni, A Megrant, P O'Malley, D Sank, A Vainsencher, J Wenner, T White, Y Yin, A N Cleland, and John M Martinis. Computing prime factors with a Josephson phase qubit quantum processor. *Nature Physics*, 8(10) :719–723, 2012.
- [6] « Stratégie nationale sur les technologies quantiques », allocution présidentielle (Emmanuel Macron), Saclay, 21 janvier 2021.

- [7] Enrique Martín-López, Anthony Laing, Thomas Lawson, Roberto Alvarez, Xiao-Qi Zhou, and Jeremy L O'Brien. Experimental realization of Shor's quantum factoring algorithm using qubit recycling. *Nature Photonics*, 6(11) :773–776, 2012.
- [8] Nissim Ofek, Andrei Petrenko, Reinier Heeres, Philip Reinhold, Zaki Leghtas, Brian Vlastakis, Ye-han Liu, Luigi Frunzio, S M Girvin, L Jiang, Mazyar Mirrahimi, M H Devoret, and R J Schoelkopf. Extending the lifetime of a quantum bit with error correction in superconducting circuits. *Nature*, 536(7617) :441–445, 2016.
- [9] Peter W Shor. Algorithms for quantum computation : discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994.
- [10] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52 :R2493–R2496, Oct 1995.
- [11] A. M. Steane. Error correcting codes in quantum theory. *Phys. Rev. Lett.*, 77 :793–797, Jul 1996.
- [12] Lieven M K Vandersypen, Matthias Steffen, Gregory Breyta, Costantino S Yannoni, Mark H Sherwood, and Isaac L Chuang. Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance. *Nature*, 414(6866) :883–887, 2001.



Les ordinateurs quantiques : un enthousiasme justifié

Giuseppe Di Molfetta¹

Church, Feynman et la suprématie quantique

L'un des débats les plus intéressants de l'histoire de l'informatique moderne s'est développé autour de la thèse de Church-Turing, qui est devenue l'un des piliers de la théorie de la calculabilité. Dans sa version physique, elle stipule que toute fonction qui peut être calculée par un système physique peut être calculée par une machine de Turing. Il ne s'agit pas d'un théorème mathématique, mais plutôt d'une proposition sur le monde « réel », sur les limites précises entre la physique et ce qui est calculable. Depuis quatre-vingts ans que Church a proposé sa thèse, personne n'a découvert de contre-exemple. Si cette thèse est communément tenue pour vraie, sa version « forte » (ou efficace) ne l'est pas [4] :

« Just as the theory of computability has its foundations in the Church-Turing thesis, computational complexity theory rests upon a modern strengthening of this thesis, which asserts that any "reasonable" model of computation can be efficiently simulated on a probabilistic Turing machine (an efficient simulation is one whose running time is bounded by some polynomial in the running time of the simulated machine). Here, we take reasonable to mean in principle physically realizable. »

1. Maître de conférences, université Aix-Marseille.

Au début des années 1980, lorsque la théorie quantique était suffisamment avancée, il est devenu évident que cette histoire devait être révisée car, pour citer Richard Feynman (« *Nature isn't classical, dammit!* »). Simuler efficacement un système régi par les lois de la mécanique quantique sur une machine de Turing classique est un problème difficile, et très probablement impossible sans une baisse exponentielle de ses performances. Le mot « probablement » est obligatoire, car il n'existe, à ce jour, aucune preuve rigoureuse de l'existence d'une séparation entre la classe *bounded-error quantum polynomial* (BQP) et la classe *bounded-error probabilistic polynomial* (BPP), les classes de complexité des problèmes efficacement résolubles sur des ordinateurs quantiques et classiques respectivement. Ce que nous savons jusqu'à présent, c'est qu'il existe des exemples dans lesquels une machine quantique garantit « seulement » une amélioration polynomiale (et c'est le cas de l'algorithme de recherche quantique introduit par Grover) et d'autres cas pour lesquels un temps de calcul polynomial est garanti là où une machine classique prendrait un temps exponentiel (par exemple dans les problèmes de factorisation ou les problèmes HSP).

Face à l'absence de preuve rigoureuse, John Preskill a inventé le terme de suprématie quantique (*quantum supremacy*) en 2012 [11], faisant écho à une brillante intuition du scientifique soviétique Yuri Manin en 1980 [10]. Ce terme indique généralement la preuve de l'existence d'un seuil au-delà duquel aucun super-ordinateur classique n'est capable de gérer la croissance exponentielle de la mémoire et de la bande passante nécessaire pour simuler son équivalent quantique. Si le terme de suprématie quantique reste aujourd'hui encore difficile à appréhender, avec des contours flous et une définition qui ne fait pas l'unanimité, cela n'a pas empêché une série de résultats importants et des avancées technologiques ces dernières années.

The experimenter's nightmare et l'optimisme de la volonté

Pour comprendre l'ampleur du défi, il faut se rappeler que la manipulation d'un seul système quantique implique des problèmes très difficiles d'un point de vue technique, dont les solutions ont posé des défis sans précédent aux scientifiques. L'idée que l'information puisse être stockée dans des états quantiques microscopiques a ouvert la perspective d'utiliser la matière quantique elle-même pour effectuer des calculs. Malheureusement, la cohérence quantique qui permettrait, par exemple, aux algorithmes quantiques les plus récents de fonctionner est très fragile. Le scepticisme omniprésent n'a pas attendu pour se faire sentir, et ce sont les extraordinaires Serge Haroche et Jean-Michel Raimond qui ont émis les premiers doutes sur la faisabilité des ordinateurs quantiques [9]; c'était en 1996, il y a vingt-cinq ans :

«... we think it fair to say that, unless some unforeseen new physics is discovered, the implementation of error correcting codes will become exceedingly difficult as soon as one has to deal with more than a few gates. In this sense

the large-scale quantum machine, though it may be the computer scientist's dream, is the experimenter's nightmare. »

Aujourd'hui, le prix Nobel Serge Haroche est l'un des protagonistes de la deuxième révolution quantique, même s'il reste prudent. Mais avant que les scientifiques puissent accepter une telle révolution, un certain nombre d'étapes — d'une importance fondamentale pour le développement d'un processeur quantique — ont dû être franchies dans le monde de la physique atomique. Les premiers pas dans cette direction ont été faits par le physicien Hans Dehmelt (prix Nobel en 1989) [6] qui a réussi à isoler un seul ion dans une chambre à vide et à le suspendre dans le vide à un point prédéterminé et contrôlable. Puis Juan Zoller et Peter Cirac ont réalisé qu'un seul ion pouvait agir comme une porte quantique [5], en construisant le premier registre quantique avec deux types distincts d'informations stockées en fonction des caractéristiques physiques de l'ion. Le premier bit quantique était né !

Deux ans après les propos de Haroche et Raimond, l'algorithme de recherche quantique, proposé par Lev Grover aux Bell Labs, a été mis en œuvre sur le premier ordinateur quantique de 3 bits (*3-qubit NMR computer*) [13]. Depuis lors, la course au développement de machines de calcul quantique de plus en plus contrôlables et programmables n'a jamais cessé, ouvrant la voie au développement de technologies considérées comme irréalisables dix ans plus tôt seulement : communications quantiques par satellite, premiers prototypes d'architectures quantiques distribuées, simulateurs quantiques programmables à 256 qubits pour étudier de nouveaux états de la matière, pour ne citer que trois des nouveaux résultats expérimentaux de cette année. Les organismes publics nationaux et internationaux et les grandes entreprises internationales telles qu'IBM et Google ont trouvé leur place dans cette histoire, et sont devenus des acteurs incontournables du futur proche des technologies quantiques.

La suprématie quantique ?

En 2019, il revient en effet à Google AI Quantum de rouvrir le débat controversé sur la suprématie quantique [2]. Après de nombreuses recherches théoriques et technologiques, les ingénieurs quantiques de Google ont construit un ordinateur quantique programmable appelé Sycamore, doté de 53 qubits et de portes quantiques avec deux qubits pour chaque paire. L'expérience informatique, très simple en soi, consistait à effectuer une mesure de tous les qubits après seulement 20 passages du circuit (problème *Random Circuit Sampling*, RCS), à répéter l'expérience quelques millions de fois pour obtenir une statistique décente, et enfin à comparer le résultat expérimental avec le résultat théorique attendu. Le même calcul, qui prenait un peu plus de deux minutes sur Sycamore, était exponentiellement plus lent sur le plus puissant ordinateur classique disponible, par exemple le Summit. Pour vous donner une idée, considérez que le Summit occupe l'espace de deux courts de tennis entiers et consomme des mégawatts d'énergie électrique, alors que Sycamore est une simple

puce dans un refroidisseur avancé. Chaque fois que nous ajoutons un seul qubit à Sycamore, Summit doit doubler de taille pour suivre. Si Sycamore avait, disons, 70 qubits, Summit occuperait l'espace d'une ville entière pour effectuer le même calcul ! Le résultat de Google n'est ni le premier ni le dernier à étonner la communauté scientifique et le grand public. Quelques mois plus tard, les chinois ont révélé leur Jiuzhang, basé sur les photons, qui a atteint une puissance de traitement de 76 qubits, estimée à des milliards de fois plus rapide que Sycamore. Le test de Google présente sans aucun doute de nombreuses limites, à commencer par les niveaux d'erreur élevés, et une preuve définitive de la suprématie quantique n'est pas claire, mais nous sommes ici confrontés au fait que, pour la première fois, un dispositif de 53 qubits a été véritablement capable de traiter 9 milliards (10^{15}) d'amplitudes de probabilités complexes, avec un avantage temporel sur n'importe quel superordinateur classique de la planète : un résultat objectivement surprenant. Il est difficile d'imaginer que ce sera la dernière, et selon les mots de l'enthousiaste Scott Aaronson [1] :

« The computer revolution was enabled, in large part, by a single invention : the transistor. Before transistors, we were stuck with failure-prone vacuum tubes. Yet vacuum tubes kind of, sort of worked : they translated abstract Boolean logic into electrical signals reliably enough to be useful. We don't yet have the quantum computing version of the transistor — that would be quantum error correction. Getting there will surely require immense engineering, and probably further insights as well. In the meantime, though, the significance of Google's quantum supremacy demonstration is this : after a quarter century of effort, we are now, finally, in the early vacuum tube era of quantum computing. »

En effet, je pense qu'il est difficile de nier, même pour les plus sceptiques, que les avancées de ces dernières années suggèrent combien et quelles étapes importantes ont été franchies dans la direction d'un hardware quantique fonctionnant dans un régime difficile à atteindre pour un ordinateur classique. Mais le scepticisme n'est pas seulement du côté des physiciens expérimentaux. De nombreux informaticiens le sont, et c'est compréhensible. D'une part, parce que les ordinateurs quantiques actuels ne permettent pas d'effectuer des calculs très complexes, en raison de l'absence de codes permettant de corriger les erreurs induites par l'environnement. En fait, de tels codes nécessiteraient des milliers de qubits physiques, un objectif qui, dans les prévisions les plus optimistes, ne sera pas atteint avant les dix prochaines années. Un autre élément qui dérange une bonne partie de la communauté informatique est le manque d'indépendance vis-à-vis du matériel physique et la mesure dans laquelle cela conditionne entièrement les modèles et les algorithmes de l'informatique quantique. Alors que l'informatique classique est rapidement parvenue à être indépendante de la machine physique, cela semble difficile, voire impossible, pour

l'informatique quantique. Une collaboration étroite entre les communautés des informaticiens et des physiciens est désormais nécessaire. En effet, si avec l'avènement des ordinateurs modernes, nous avons assisté à une séparation progressive entre les premiers et les seconds, l'informatique quantique soulève de nouvelles questions qui se situent intimement à la frontière entre la physique et l'informatique théorique. C'est dans cette zone frontière de plus en plus vaste que de nombreux étudiants ont fait leurs premiers pas dans la recherche, une dynamique qui, à terme, risque inévitablement de pénaliser le système d'enseignement et de recherche français, encore peu interdisciplinaire.

Le calcul quantique : une nouvelle grammaire

L'enthousiasme croissant suscité par ces questions n'est pas seulement lié au développement des premiers ordinateurs quantiques, qui suscite en soi l'intérêt d'un vaste réseau d'acteurs privés et publics. Cet enthousiasme a des racines plus profondes : aujourd'hui, les modèles informatiques conçus par les informaticiens deviennent la nouvelle grammaire pour l'étude des processus naturels. Après avoir conquis la mécanique quantique et la thermodynamique, l'information et son traitement sont devenus centraux, par exemple, dans l'étude de la gravité ou de la biologie. Dans cette révolution, informaticiens et physiciens travaillent et travailleront ensemble, pour développer un langage commun, trente ans après les mots de Feynman.

De la simulation de la nature...

L'un des domaines dans lesquels cette symbiose semble être la plus forte est la simulation quantique, l'une des principales applications à court terme des ordinateurs quantiques. L'idée d'un simulateur quantique trouve son origine dans les premiers travaux de David Deutsch et de sa machine de Turing quantique universelle [7] qui représente dans la théorie de la calculabilité quantique exactement ce que la machine de Turing universelle représente pour la calculabilité classique. L'idée d'une machine quantique universelle a ensuite conduit Seth Lloyd, en 1996, à démontrer qu'une telle machine agit effectivement comme un simulateur quantique universel. Pouvoir simuler un phénomène physique dont la théorie est connue est d'un intérêt extraordinaire, car cela nous donne accès à l'étude d'états exotiques de la matière auxquels nous n'aurions pas accès autrement, et inspirera probablement de nouvelles générations d'ordinateurs quantiques. Mais il est encore plus intéressant de comprendre dans quelle mesure un simulateur quantique peut réellement capturer la réalité qui nous entoure. Cette question soulève des interrogations profondes sur la version quantique de la thèse de Church-Turing, et peut en même temps suggérer de nouveaux problèmes dans la compréhension de la nature.

Par exemple, nous savons maintenant que les physiciens utilisent ce qu'ils appellent la théorie des champs quantiques pour décrire les particules élémentaires. Or, formellement, une telle théorie nécessite un nombre infini de degrés de liberté par unité de volume. Prétendre simuler exactement une telle théorie à l'aide d'une machine à états finis est tout simplement impossible. Par définition, un simulateur quantique nécessite une description discrète du phénomène à simuler, où discret signifie un système constitué de composants disjoints, par exemple des qubits. La question de savoir si un circuit quantique peut saisir toute la réalité d'un phénomène physique à une telle échelle reste ouverte et continuera sans aucun doute à faire l'objet d'études dans les années à venir. Autre exemple, le problème de la combinaison d'un phénomène quantique discret avec le concept d'espace-temps continu est l'énigme la plus fascinante dans la communauté des physiciens depuis des décennies. La formulation d'un modèle de gravité quantique a soulevé des questions fondamentales sur la causalité, la localité, l'unitarité, et l'osmose entre la physique et l'informatique a rendu possible des descriptions alternatives, de nouveaux langages. Le bagage théorique des informaticiens pourrait jeter un nouvel éclairage sur ce grand défi, en faisant entrer des concepts tels que l'universalité, la classification et la complexité dans une zone frontalière toujours plus riche, ouvrant de nouvelles perspectives à la recherche de réponses et, en définitive, au progrès scientifique. Mais il existe bien d'autres domaines dans lesquels ces simulateurs quantiques de quelques dizaines de qubits jouent un rôle novateur, de l'étude des différentes phases quantiques de la matière, cruciale pour la conception de nouveaux matériaux, aux premières simulations de réactions chimiques, comme la dernière réalisée sur Sycamore il y a tout juste un an. Dans cette nouvelle expérience, l'équipe AI Quantum a concentré ses efforts sur la simulation d'un processus chimique simple — l'approximation Hartree-Fock d'un système chimique réel — dans ce cas particulier, une molécule de diazène réagissant avec des atomes d'hydrogène [3]. Bien sûr, une telle simulation ne dit encore rien de la supériorité des ordinateurs quantiques, puisque la même simulation peut être réalisée efficacement sur un ordinateur classique. Mais elle confirme la validité méthodologique, la preuve de principe, des éléments de base de ce qui, dans dix ans, conduira à des simulations chimiques d'un ou deux ordres de grandeur plus importants [14].

L'intérêt de ces récents résultats, pour revenir au cauchemar des expérimentateurs, est que nous n'avons pas eu besoin d'attendre des ordinateurs quantiques parfaits et tolérants aux pannes. L'avènement de machines telles que Sycamore et Jiuzhang, communément appelées NISQ (*noisy intermediate-scale quantum*), sont déjà capables d'innover dans des régimes qui seront bientôt difficiles à atteindre pour les ordinateurs classiques actuels, en simulant un large éventail de systèmes quantiques existant dans la nature, bien qu'en utilisant un nombre limité de ressources.

... à la nature du calcul

Les simulateurs quantiques ont joué un rôle fondamental pour repenser certains phénomènes physiques. Nous en avons déjà parlé à plusieurs reprises. Ce dont nous n'avons pas suffisamment parlé, c'est de la mesure dans laquelle un phénomène naturel peut inspirer de nouveaux modèles ou algorithmes de calcul. De nombreux modèles classiques non conventionnels de calcul sont inspirés par des phénomènes naturels, par exemple en biologie, on peut citer les membranes cellulaires ou le système immunitaire. Certains de ces phénomènes ont inspiré de nouvelles familles d'algorithmes, comme dans le cas des fourmis : des insectes sociaux hautement coopératifs qui recherchent de la nourriture en grands groupes et communiquent indirectement à l'aide d'une substance chimique appelée phéromone. Fait remarquable, chaque fourmi explore l'espace de solution d'un problème et communique avec ses congénères par le biais d'une variable partagée, à savoir la phéromone. Ce comportement a permis le développement d'une nouvelle famille d'algorithmes d'optimisation et de recherche (*Ant Colony Optimization* [8]). Parfois, certains de ces phénomènes ont le mérite d'être facilement accessibles en nature, plus rarement programmables et bénéficient certainement de millions d'années d'adaptation en avance sur l'informatique à peine plus qu'adolescente. Non seulement la biologie, mais les particules élémentaires pourraient mettre en œuvre certains algorithmes que nous connaissons d'une manière différente et peut-être en suggérer de nouveaux. Récemment, des chercheurs français [12] ont montré que les défauts topologiques dans les matériaux cristallins pourraient se comporter comme des oracles dans un algorithme de recherche, et que les électrons qui s'y propagent « chercheraient » ces défauts avec la même complexité que le célèbre algorithme introduit par Grover. Mais qu'est-ce que cela a à voir avec la course actuelle aux technologies quantiques ? En première approximation, pas beaucoup, et de telles preuves pourraient inspirer, dans le meilleur des cas, de nouvelles familles d'algorithmes non conventionnels, comme l'algorithme de nos chères fourmis. Le fait est que les algorithmes quantiques, tels que nous les connaissons, nécessitent des ordinateurs quantiques suffisamment tolérants aux erreurs, et un tel horizon est encore loin des possibilités techniques actuelles. Que certains algorithmes quantiques se réalisent spontanément dans la nature dans des conditions environnementales défavorables pourrait constituer un raccourci fascinant pour une nouvelle génération d'ordinateurs quantiques (analogiques) « naturels ». Et malgré la prudence qui s'impose à l'égard de ce type de résultats et de leur impact potentiel, il reste tout à fait d'actualité de comprendre dans quelle mesure les progrès de la simulation quantique de phénomènes physiques jusqu'alors difficiles d'accès peuvent non seulement nous aider à appréhender la nouvelle physique, mais aussi avoir des retombées intéressantes et inattendues dans le développement de nouvelles technologies.

Références

- [1] Scott Aaranson, NYT, 2019.
- [2] Arute, Frank, et al. *Quantum supremacy using a programmable superconducting processor*, Nature, 574.7779, 2019.
- [3] Arute, Frank, et al. *Hartree-Fock on a superconducting qubit quantum computer*, Science, 369.6507, 2020.
- [4] Bernstein, E. et Vazirani, U. *Quantum complexity theory*, SIAM Journal on Computing, 26, 5 octobre 1997.
- [5] Cirac, Juan I., et Peter Zoller. *Quantum computations with cold trapped ions*, Physical review letters, 1995.
- [6] Dehmelt, Hans. *A single atomic particle forever floating at rest in free space : New value for electron radius*, Physica Scripta, T22, 1988.
- [7] Deutsch, David. *Quantum theory, the Church–Turing principle and the universal quantum computer*, proceedings of the Royal Society of London, A. Mathematical and Physical Sciences, 400.1818, 1985.
- [8] M. Dorigo & T. Stützle, Ant Colony Optimization, MIT Press, 2004.
- [9] Serge Haroche et Jean-Michel Raimond, *Physics Today*, 1996.
- [10] Manin, Yuri, *Computable and uncomputable* (in russian), Sovetskoye Radio, Moscow, 1980.
- [11] Preskill, John. *Quantum computing in the NISQ era and beyond*, Quantum, 2018.
- [12] Roget, Mathieu, et al. *Grover search as a naturally occurring phenomenon*, Physical Review Letters, 124.18, 2020.
- [13] Vandersypen, Lieven MK, et al. *First implementation of a three quantum bit search algorithm*. No. quant-ph/9910075, 1999.
- [14] Yuan, Xiao. *A quantum-computing advantage for chemistry*, Science, 369.6507, 2020.



Écosystème quantique en France et dans le monde

Olivier Ezratty¹

L'informatique quantique, au sens large, est un domaine technologique stratégique pour diverses raisons. En cryptographie, la souveraineté des États est en jeu avec la protection des communications sensibles. L'informatique quantique a des applications critiques et pourrait étendre le champ du possible au-delà de ce qui est réalisable aujourd'hui en informatique classique, notamment dans les domaines de la santé, de l'environnement et de l'intelligence artificielle.

En termes de maturité, les cryptographies quantique et post-quantique représentent des domaines plus établis avec des acteurs économiques et des solutions commerciales, même si la normalisation de la cryptographie post-quantique n'est pas encore achevée. Cependant, elle comporte moins d'inconnues scientifiques et techniques que l'informatique quantique.

La faisabilité d'ordinateurs quantiques utiles sur le plan commercial reste une question ouverte. D'importants défis technologiques sont à relever, notamment l'épineuse question du bruit des qubits, de la correction d'erreurs quantiques et des technologies permettant d'augmenter le nombre de qubits physiques de plusieurs ordres de grandeur. L'informatique quantique est donc pleine d'incertitudes scientifiques et technologiques, avant même d'être économiques et commerciales.

Pour l'instant, la recherche est principalement financée par les gouvernements dans la plupart des pays, puis par de très grands acteurs informatiques qui font de

1. Consultant et auteur, notamment des ouvrages de référence « Comprendre l'informatique quantique » (septembre 2020) et sa version actualisée en anglais *Understanding Quantum Technologies* (septembre 2021), diffusés gratuitement en PDF. Il anime aussi deux séries de podcast (*Quantum et Decode Quantum*) avec Fanny Bouton, olivier@oezratty.net.

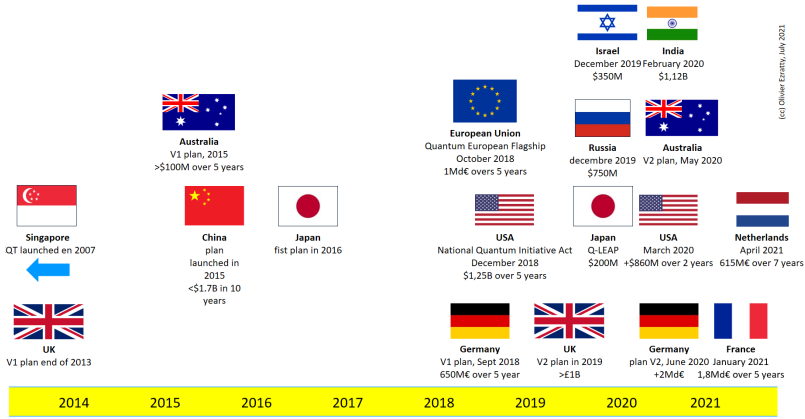












FIGURE 1. Investissements mondiaux en informatique quantique depuis 2014.

nombreux paris technologiques en parallèle (Google , Intel , Microsoft , IBM , Alibaba), et enfin par quelques startups plus ou moins bien financées, principalement en Amérique du Nord (D-Wave, IonQ , Rigetti, PsiQuantum) et, de plus en plus, en Europe (IQM, Pasqal, C12).

Le domaine des logiciels pour ordinateurs quantiques n'en est pour sa part qu'à ses débuts. Les principaux acteurs et startups créant des ordinateurs quantiques ont tous investi dans les logiciels, en commençant par les outils de développement et la création d'algorithmes quantiques valorisant leur matériel. Chacun vise évidemment à créer des plateformes logicielles de premier plan dont certaines sont déjà disponibles dans le cloud, comme celles d'IBM, d'Amazon, de Microsoft et de D-Wave. La plupart de ces acteurs, ainsi qu'Atos, offrent un accès en cloud à des émulateurs quantiques tournant sur des ordinateurs classiques.

Investissements mondiaux

Les États-Unis et la Chine sont évidemment en tête des investissements publics dans les technologies quantiques (cf. figure 4), avec une bonne part qui finance la recherche fondamentale. La répartition de ces investissements, qui comprennent à la fois la cryptographie quantique, les ordinateurs quantiques et les capteurs quantiques, intrigue les autres pays. Les technologies quantiques sont devenues un enjeu géopolitique, presque comme la dissuasion nucléaire. Les pouvoirs publics de ces différents pays se sont mobilisés de manières très différentes sur les technologies quantiques.

	COUNTRY	TP	%TP	TC	%TC	CPP	RCI	%ICPEI
1	 USA	4,295	26.4%	108,128	44.8%	25.2	1.7	70%
2	 China	3,706	22.8%	38,611	16.0%	10.4	0.7	44%
3	 UK	1,428	8.8%	32,435	13.4%	22.7	1.5	120%
4	 Germany	1,400	8.6%	38,339	15.9%	27.4	1.9	123%
5	 Japan	1,106	6.8%	20,996	8.7%	19.0	1.3	99%
6	 Canada	1,056	6.5%	23,104	9.6%	21.9	1.5	124%
7	 India	991	6.1%	5,847	2.4%	5.9	0.4	33%
8	 Australia	777	4.8%	20,777	8.6%	26.7	1.8	130%
9	 France	699	4.3%	14,016	5.8%	20.1	1.4	117%
10	 Italy	635	3.9%	10,522	4.4%	16.6	1.1	116%
Total 10 countries		16,093	98.9%	312,775	129,5%	19.4	1.3	83.1%
Total world		16,279		241,536		14.8		

*TP= Total Publication ; TC = Total Citation ; CPP = Citation par Publication = TC/TP ;
RCI = Relative Citation Index ; ICPEI = International Collaboration Publication Extended Index

FIGURE 2. Montant des investissements mondiaux par pays.

La plupart des gouvernements des pays développés ont coordonné les efforts dans le domaine quantique avec des plans allant jusqu'à deux milliards de dollars sur 5 ou 10 ans. Il est cependant difficile de comparer ces investissements entre pays. En cause, la prise en compte ou l'absence de prise en compte des investissements existants, les investissements non divulgués dans l'armée et le renseignement, et l'éventuel embellissement de la mariée comme c'est le cas avec la Chine qui communique souvent des montants agrégeant les technologies quantiques et d'autres technologies comme l'intelligence artificielle.

Une évaluation des publications scientifiques en informatique quantique réalisée par des étudiants de l'Insead en 2018 a montré sans surprise que les USA, le Canada et la Chine sont les premiers pays à publier². Une analyse plus détaillée a été produite par Michel Kurek en septembre 2020 (cf. figure 2) qui a permis de relativiser l'influence des publications chinoises³. En effet, le nombre de citations par publications est très faible en Chine et en Inde, comparé à tous les pays occidentaux. Les

2. <https://www.insead.edu/sites/default/files/assets/dept/centres/gpei/docs/insead-student-quantum-computing-investment-analysis-apr-2018.pdf>.

3. https://www.researchgate.net/publication/350453567_Quantum_Technologies_Patents_Publications_Investissements_Landscape.

investissements importants réalisés par les pays développés dans les technologies quantiques font craindre que la puissance de calcul ne finisse par se concentrer entre les mains de quelques-uns, voire d'un seul pays ou d'une seule entreprise. Je ne le crois pas, du moins pas dans la phase initiale de développement de ces technologies. Les connaissances sur le sujet sont très distribuées, tout comme les technologies habilitantes et les matériaux stratégiques. Je situerais plutôt le risque de concentration dans une deuxième phase de maturation de ce marché, qui verra un marché initialement fragmenté avec de nombreux acteurs se concentrer par consolidation comme c'est généralement le cas dans le numérique. Elle le fera probablement pour des raisons plus macro-économiques que scientifiques ou technologiques, par des économies d'échelle et la « plateformesisation » des offres. Cela explique pourquoi il est nécessaire de surveiller simultanément le matériel, les outils de développement et les applications logicielles de l'informatique quantique.

Une fois les principales hypothèses scientifiques et technologiques levées, le succès de chaque entreprise et pays dépendra des facteurs clés de succès classiques des écosystèmes technologiques : rapidité d'exécution, qualité des équipes, niveau de financement, marketing, communication, vente et promotion des plateformes technologiques adoptées par un maximum d'acteurs et à l'échelle mondiale. C'est là que des approches souverainistes combinant le protectionnisme des acteurs clés tout en assurant une ouverture commerciale maximale sur le monde pour leur permettre de réaliser des économies d'échelle devront être soigneusement adaptées.

La figure 3 montre quels pays maîtrisent le mieux les différentes technologies de calcul quantique par type de qubit. Globalement, nous avons un bon équilibre entre les États-Unis et l'Union européenne, même si les États-Unis ont l'avantage d'avoir de grands fournisseurs informatiques investis dans le domaine des supraconducteurs (Google, IBM), du silicium (Intel), des ions piégés (Honeywell) et des qubits topologiques (Microsoft).

Quels sont les principaux indicateurs de réussite des pays qui investissent dans la course quantique ?

Des cabinets d'analystes vont probablement créer leur propre classements quantiques « de Shanghai » en utilisant des mesures composites : financement public, publications scientifiques, brevets et autres, esprit d'entreprise, nombre de startups, financement des startups, investissements des grandes entreprises, adoption par les entreprises, main-d'œuvre qualifiée, etc. Devinez quoi ? Les États-Unis et la Chine seront probablement classés premiers. Et les petits pays derrière dans un ordre variable. Mais que se passerait-il si l'Europe était consolidée ? Elle serait probablement positionnée entre les USA et la Chine.

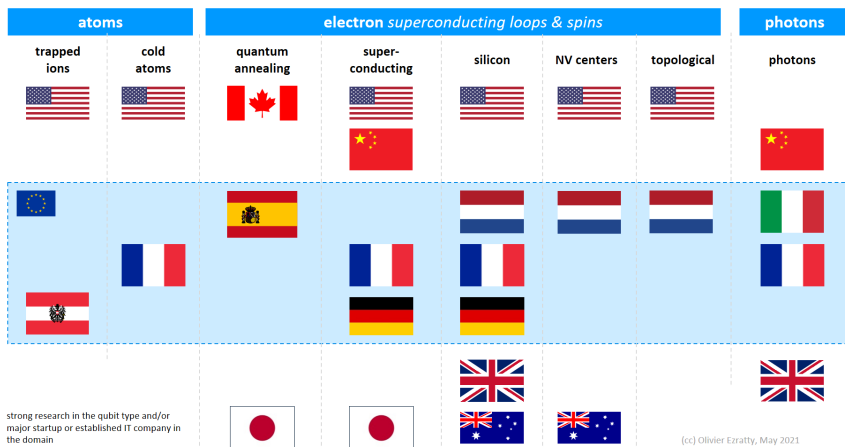


FIGURE 3. Maîtrise des technologies par les pays.

Écosystème entrepreneurial

Les entreprises des technologies quantiques sont à la fois des startups, des PME établies et de grandes entreprises du numérique qui s’y sont lancées dans une approche de diversification.

La cartographie de ces acteurs est un peu plus facile que pour d’autres technologies telles que l’intelligence artificielle, car ils ne sont pas encore très nombreux. On en dénombre aujourd’hui plus de 450 dans le monde avec un peu moins de trois milliards de dollars de levées de fonds depuis leur création ce qui est, somme toute, modeste.

Cet écosystème a commencé à prendre forme avant même que les ordinateurs quantiques ne fonctionnent à petite échelle. Il est fascinant de découvrir des startups qui font des paris à long terme, en particulier dans le domaine du matériel. Les startups logicielles s’appuient sur une infrastructure matérielle encore limitée mais réduisent souvent leurs risques en prenant également en charge les architectures informatiques traditionnelles. Leurs clients sont de grandes entreprises qui testent des algorithmes à petite échelle, souvent sur des calculateurs à recuit quantique de D-Wave et parfois avec IBM qui est très pressant dans ses efforts d’évangélisation quantique. À ce jour, aucune application ne semble avoir été déployée en production. Nous sommes donc dans le domaine de la recherche appliquée et du prototypage à petite échelle au sein des entreprises utilisatrices.

De leur côté, les systèmes de cryptographie quantique sont opérationnels et correspondent à un marché bien distinct, tout comme le marché très fragmenté des capteurs quantiques.

Les startups sont pour la plupart américaines et européennes. L'écosystème logiciel est à suivre de près. Il se développera probablement lorsque le matériel fonctionnera à plus grande échelle, notamment avec les ordinateurs bruités de première génération (*Noisy Intermediate Scale Quantum Computers* ou NISQ) et les simulateurs quantiques.

Les enjeux de nombreuses startups dans ce domaine sont communs avec ceux des deep techs : comment développer de vrais produits avec des économies d'échelle, comment se développer rapidement à l'international, comment ne pas tomber dans des modèles trop « serviciels », et comment résister à ce que certains appellent déjà l'hiver quantique, une période de perte d'intérêt pour le sujet que certains anticipaient en 2020 mais qui n'a finalement pas encore eu lieu. Dans le créneau des technologies habilitantes (sources de photons, cryostats, ultravide, divers capteurs, électronique), les startups et PME concernées s'en sortent bien en touchant des marchés diversifiés, ciblant notamment plusieurs branches différentes de la recherche ou des applications militaires ou aérospatiales.

Investisseurs et startups

Les premiers fonds d'investissement plus ou moins spécialisés dans les technologies quantiques ont déjà vu le jour. Le plus connu est français ! Quantonation est un fonds d'amorçage français créé par Charles Beigbeder et géré par Christophe Jurczak, un physicien et ancien thésard d'Alain Aspect passé ensuite par le secteur privé, notamment aux USA. Ils ont déjà investi dans une douzaine de startups dont Pasqal (France, calculateurs quantiques à base d'atomes froids), Quantum Benchmark (Canada, logiciel), Kets Quantum Security (UK, composants dans la cryptographie quantique), Orca Computing (UK, calcul photonique), CryptoNext Security (France, logiciels de cryptographie post-quantique), Qunnect (États-Unis, répéteurs pour QKD), Quandela (France, source de photons uniques), Qubit Pharma (simulation moléculaire), Qnami (Suisse, métrologie quantique exploitant des centres NV, aussi appelées cavités dans des diamants), Orca Computing (Royaume-Uni, qubits photoniques) et Qphox (Pays-Bas, communications entre ordinateurs quantiques). Il est aujourd'hui le principal animateur de l'écosystème des entreprises quantiques en France avec le Lab Quantique, qui a été créé conjointement avec Bpifrance. Le fait d'avoir un fonds d'amorçage français qui investit aussi à l'international est un véritable atout pour la France.

Quantum Valley Investments (QVI) est un fonds d'investissement canadien de 100 millions de dollars, levé en 2013, dédié aux technologies quantiques. Leurs fondateurs avaient investi en 1984 dans Blackberry/RIM. Ils ne divulguent pas leurs



FIGURE 4. Répartition des investissements par pays.

investissements, sauf dans ISARA Corporation, dont une partie sont des *spin-offs* du laboratoire de recherche canadien *Institute for Quantum Computing* de l'université de Waterloo en Ontario.

D'autres fonds investissent dans les technologies quantiques mais sont moins spécialisés : Quantum Ventures (USA), Summer Capital (Pays-Bas) qui a investi dans Horizon Quantum Computing, Rigetti et Turing, Phystech Ventures, créé par des Russes dans la Silicon Valley et ayant déjà investi dans IDQ et Nano-Meta, Parkwalk Advisors (UK) qui a investi dans Phasecraft, Quantum Motion Technologies, Riverlane, nu quantum, nu nano et Oxford Quantum Circuits et SpeedInvest (Autriche) qui a investi dans QPhox (Pays-Bas) et Kets (UK).

Les investissements dans les startups ont commencé à décoller dans le monde entier à partir de 2016⁴. Bon nombre des startups du domaine ne sont pas encore dans la forme « pure » du modèle startupien, c'est-à-dire qu'elles sont loin d'avoir un modèle ou même, juste un produit. Il s'agit souvent soit de petites entreprises industrielles ciblant des marchés de niche à très faible volume, soit de startups où le risque scientifique et technologique est encore très élevé avant de pouvoir vendre quoi que ce soit. C'est souvent une combinaison des deux. Elles peuvent alors se financer avec de la recherche contractuelle et divers services de conseil pour de grandes entreprises ou des institutions publiques, en plus de levées de fonds capitalistiques classiques.

Sur la figure 5, on peut voir le nombre de créations de startups par an et sur la partie gauche de la figure 6 une ventilation par pays.

Le diagramme de droite de la figure 6 contient une répartition différente par pays qui met en évidence les financements les plus importants. Comme d'habitude, nous constatons un écart de financement important entre l'Amérique du Nord et l'Europe. L'une des raisons est que les startups européennes ont été créées plus tard⁵. L'autre raison est bien sûr un accès différent au capital. Et 70 % du financement mondial

4. <https://medium.com/speedinvest/european-seed-investment-quantum-applications-aalf107a76ee>.

5. <https://medium.com/le-lab-quantique/new-record-looms-in-vc-funding-of-quantum-startups-484b82541004> et <https://medium.com/uvc-partners-news/the-european-quantum-computing-startup-landscape-a115ffe84ad8>.

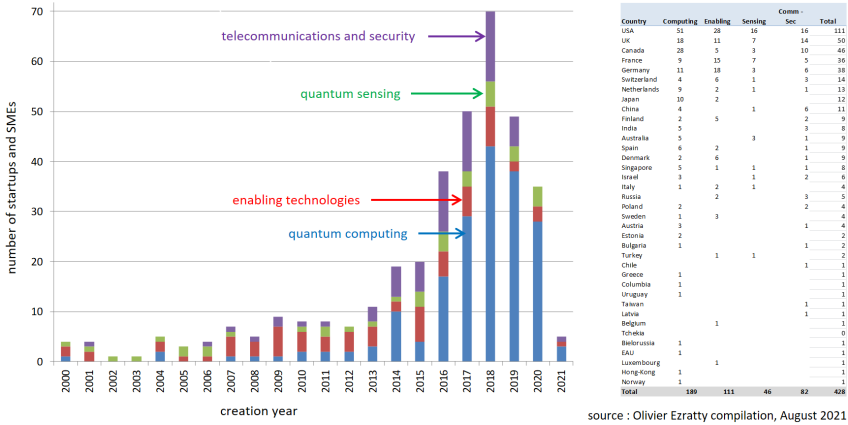


FIGURE 5. Nombre de créations de startups par an.

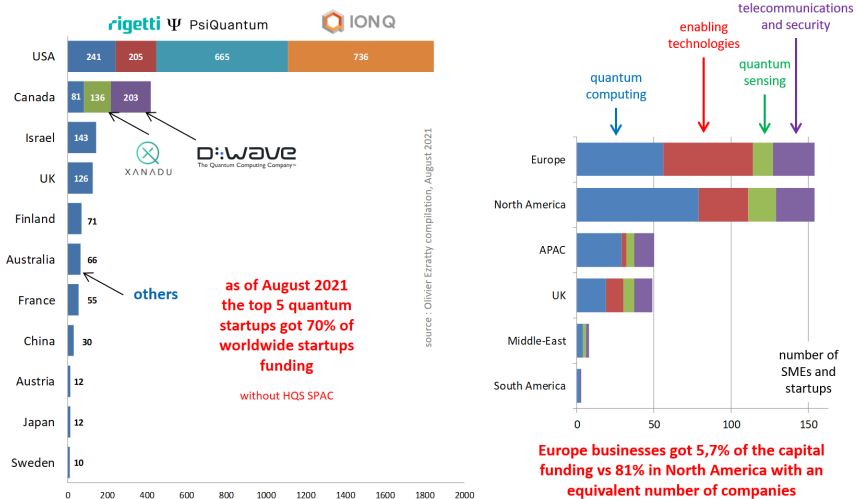


FIGURE 6. Nombre de créations de startups par an.

des startups est allé aux cinq premières startups que sont IonQ, PsiQuantum, Rigetti, D-Wave et Xanadu, toutes originaires des États-Unis et du Canada, alors que l'Europe compte un nombre équivalent d'entreprises par rapport à l'Amérique du Nord. S'il y a autant de startups et de petites entreprises quantiques en Europe continentale

qu'en Amérique du Nord, leur financement par capitaux propres visibles ne représente que 5,7 % du financement mondial, alors que les entreprises nord-américaines en ont obtenu 81 %. Et cela n'inclut pas les investissements d'IBM, Google, Intel et Honeywell.

Investissements par pays

Dans ce qui suit, nous allons faire le tour de la situation dans quelques pays. Tous ne peuvent pas être mentionnés, comme le Canada, l'Australie, l'Autriche, la Suisse, les Pays-Bas et le Japon, qui sont aussi tous très actifs dans les technologies quantiques.

USA

Comme dans tous les domaines du numérique, les USA dominent le paysage des technologies quantiques. Leur plan quantique national, le *National Quantum Initiative Act*, a été lancé fin 2018 avec un financement de 1,2 milliard de dollars sur cinq ans, complétés par une rallonge de 860 millions de dollars allouée en mars 2020 sur deux ans. Cette recherche finance les laboratoires de recherche publics et privés civils des universités via le Département de l'énergie (vague équivalent du CEA français), la *National Science Foundation* (équivalent de l'ANR) et le NIST (équivalent du LNE). Ce plan a donné lieu à la création de huit centres de recherche quantique bâtis sur l'existant des universités ou des organismes de recherche fédéraux (cf. figure 7) : trois *Quantum Leap Challenge Institutes* financés par la NSF et cinq centres de recherche intégrés dans des laboratoires du Département de l'énergie (Lawrence Livermore, Oak Ridge, Sandia Labs...). Le tout est complété par des financements non documentés liés à la défense et au renseignement.

Le pays se distingue aussi par les investissements significatifs des grands acteurs du privé que sont IBM, Google, Amazon, Rigetti dans les qubits supraconducteurs, d'IonQ et Honeywell dans les ions piégés, par un écosystème de plus d'une centaine d'entreprises de toutes tailles, par son rôle dans la standardisation de la cryptographie post-quantique via l'initiative associée du NIST lancée en 2016.

Royaume Uni

C'est le pays de l'Union Européenne (à l'époque) qui s'est engagé le premier et le plus activement dans les technologies quantiques, et ce, dès 2013, avec un plan de cinq ans renouvelé en 2019. Le pays a investi plus de 1,2 milliard de livres sterling en huit ans dans le secteur, financements privés compris. Le gouvernement britannique avait dès le départ souligné l'importance de la valorisation industrielle de la recherche. Le plan a conduit à la création de quatre hubs quantiques thématiques pilotés par des universités : Birmingham (capteurs quantiques), Glasgow (imagerie), Oxford (calcul, cryptographie) et York (communications). En 2019, le pays lançait

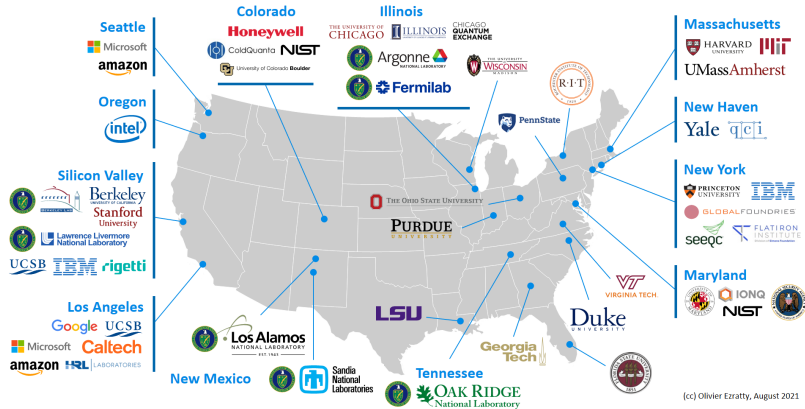


FIGURE 7. Investissements quantiques aux USA.

la création d'un centre de calcul quantique dédié, le *National Quantum Computing Center*.

Le plan a porté ses fruits, faisant du pays le second au monde par la densité de son écosystème entrepreneurial quantique. On y trouve une masse critique de startups dans les capteurs quantiques (M Squared, Cerca, Oxford HighQ), ainsi que dans la cryptographie quantique (Kets), dans les technologies de calcul quantique comme avec Orca Computing dans les photons, Oxford Quantum Technologies dans les qubits supraconducteurs, Oxford Ionics et Universal Quantum dans les ions piégés ou Quantum Motion dans les qubits à spin d'électrons. Leur tissu industriel dans les technologies habilitantes est aussi bien développé avec Oxford Instrument (cryogénie), Oxford Ionics, Kelvin, etc.

Allemagne

L'Allemagne a lancé son premier plan quantique en septembre 2018 avec 650 millions d'euros à la clé. Il a été complété par un plan de deux milliards d'euros lancé en juin 2020 dans le cadre des plans de relance économiques liés à la pandémie covid-19. Ce plan comprend le déploiement dans le pays de deux ordinateurs quantiques NISQ ainsi que celui d'un réseau gouvernemental de distribution de clés quantiques.

La recherche publique allemande est très active autour des technologies quantiques. Ils investissent dans les qubits supraconducteurs, les spins d'électrons et les ions piégés. Ce sont des Allemands qui pilotent le plus grand nombre de projets de l'*European Quantum Flagship* avec cinq projets.

Les entreprises du secteur sont plutôt focalisées sur les technologies habilitantes comme les lasers, les logiciels ainsi que la cybersécurité et les capteurs quantiques. Ils n'ont quasiment pas de startups établies dans l'informatique quantique, ce qui

explique indirectement pourquoi ils ont accueilli un laboratoire de recherche du finlandais IQM et un ordinateur quantique d'IBM installé près de Stuttgart et géré avec le concours du Fraunhofer Institute.

L'une des forces de l'Allemagne est l'engagement de ses grands industriels dans l'évaluation des technologies quantiques. Cela a même abouti à la création, en juin 2021, de *Quantum Technology and Application Consortium* (QUTAC) avec comme membres fondateurs BASF, BMW, Boehringer Ingelheim, Bosch, Infineon, Merck, SAP, Siemens et Volkswagen.

Chine

La Chine impressionne beaucoup avec ses investissements dans les *deep techs*. Son premier plan quantique gouvernemental structuré a été lancé en 2015. En pratique, le pays a investi un peu moins de deux milliards de dollars en une dizaine d'années ; ce qui est moins impressionnant que certains montants relayés comme 10 milliards ou 30 milliards de dollars qui englobaient en général d'autres secteurs technologiques que le quantique tels que celui de l'intelligence artificielle.

La recherche est notamment coordonnée par Jian-Wei Pan de l'USTC (*University of Science and Technology of China*) et de la *Chinese Science Academy* (CAS). Le pays a lancé la construction d'un laboratoire de recherche géant à Hefei, mais qui couvre aussi l'intelligence artificielle.

Du côté scientifique et technologique, le pays se distingue par un fort investissement dans la distribution de clés quantiques avec un réseau établi de 2000 km reliant Beijing à Shanghai, en cours d'extension à d'autres villes avec plus de 30 000 km en tout. Ils ont aussi testé la distribution de clés quantiques par satellite en 2017. C'est impressionnant mais n'a pas de grand intérêt pratique car le débit de la transmission est extrêmement faible, de l'ordre d'un seul bit par seconde.

Dans les qubits, les équipes de recherche chinoises ont atteint des records en photonique avec des prouesses d'échantillonnage de bosons, des expériences de mélanges aléatoires de photons qui permettent de plastronner avec des avantages quantiques qui n'en sont pas vraiment car ces systèmes ne sont pas du tout programmables.

Des équipes chinoises ont aussi publié quelques travaux intéressants dans les qubits supraconducteurs, avec des circuits de 66 qubits dont seulement 56 sont opérationnels. Mais la technologie présentée en 2021 reprenait quasiment à l'identique, sans l'améliorer, celle de Google Sycamore de 2019.

Cela correspond à une approche chinoise somme toute traditionnelle : ils étudient les travaux de recherche occidentaux, ils les reproduisent dans leurs laboratoires et les améliorent à la marge. Cela leur permet au fil de l'eau de monter en puissance. Les chercheurs chinois sont de grands déposants de brevets comme nous l'avons déjà vu mais l'influence de ces brevets semble faible à ce stade.

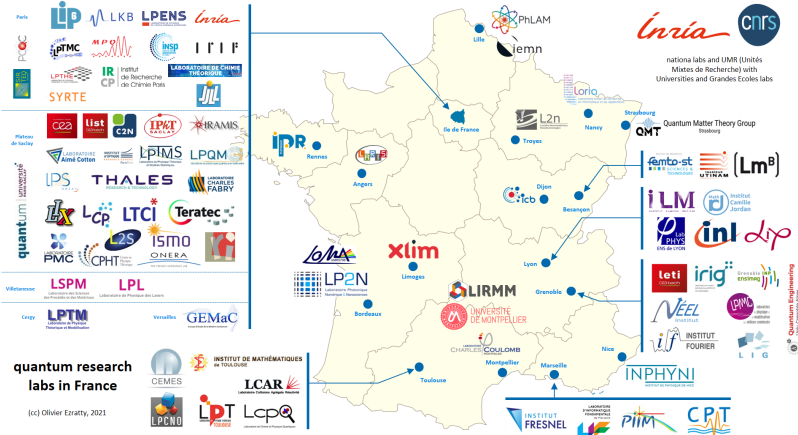


FIGURE 8. Investissements quantiques en France.

Du côté du secteur privé, les grands acteurs locaux du cloud ont lancé des offres d’émulation ou de calcul quantique, comme chez Huawei, Tencent et Baidu.

Leur écosystème de startups est relativement petit avec environ une dizaine d’entreprises identifiables, ceci s’expliquant peut-être par le confort des chercheurs dans les laboratoires qui sont bien financés.

À noter que la Chine, comme l’Asie en général, ne semble pas avoir d’influence notable sur la dimension logicielle du calcul quantique.

France

La France a lancé son plan quantique relativement en retard, en janvier 2021, avec un montant total de 1,8 milliards d’euros dont un milliard d’euros provenant de financements de l’État, le reste venant de l’Europe et du secteur privé. C’est un plan qui finance tout le cycle de l’innovation allant de la recherche fondamentale à la valorisation industrielle. L’accent est notamment mis sur les filières du calcul quantique (atomes froids, supraconducteurs à qubits de chats et spins d’électron). Le plan traite aussi bien à part les technologies habilitantes comme la cryogénie, les lasers et la cryoélectronique. Les capteurs quantiques ne sont pas en reste, tout comme le logiciel où un rattrapage rapide doit être réalisé aussi bien du côté de la recherche que de celui des startups.

La France dispose d’un large éventail d’activités de recherche et d’industrie dans le domaine des technologies quantiques (cf. figure 8). La recherche publique est organisée autour du CNRS, du CEA et d’Inria. Le CNRS fait de la recherche fondamentale en physique, en mathématiques et en algorithmique. Le CEA fait également

de la recherche fondamentale en physique, notamment sur les qubits supraconducteurs, et de la recherche appliquée sur les qubits à spins d'électrons en technologie CMOS ainsi que sur la photonique. Enfin, Inria mène des recherches en informatique quantique, sur la correction d'erreur quantique, la cryptographie et les algorithmes quantiques. De nombreux laboratoires sont des unités mixtes de recherche entre les universités et ces organismes nationaux. Les laboratoires de recherche quantiques sont concentrés en Île de France et à Grenoble, mais d'autres centres régionaux sont actifs comme Toulouse, Montpellier, Marseille, Lyon, Bordeaux, Besançon et Lille.

Du côté des startups et PME, la France compte environ 35 acteurs. Ceux qui planchent sur la création d'ordinateurs quantiques sont Alice&Bob (qubits de chat), C12 (qubits de spins d'électrons de nanotubes de carbone), Pasqal (qubits d'atomes froids) et Quandela (sources de photons uniques et qubits de photons).

Du côté des logiciels, nous avons Qubit Pharmaceuticals (simulations pour les biotechs), QuantFi (finance), VeriQloud (télécommunications quantiques) et Prevision.io (apprentissage automatique quantique), ainsi qu'une série d'entreprises spécialisées dans la cryptographie, principalement en cryptographie post-quantique avec CryptoNext, CryptoExperts, Ravel et Secure-IC. D'ailleurs, un bon nombre de scientifiques français, dont ceux de CryptoNext, ont contribué aux projets finalistes de la consultation du NIST américain sur la standardisation de la cryptographie post-quantique.

Dans le domaine des capteurs quantiques, nous avons surtout Muquans (micro-gravimètres, acquis par ixBlue en 2021) et Thales (centres NV, SQUIDS, atomes froids, cryogénie embarquée).

La grande majorité de ces startups ont été créées par des chercheurs et ont gardé des liens avec leurs laboratoires de recherche d'origine. Ainsi, Pascale Senellart, directrice de recherche au CNRS au laboratoire C2N du CNRS et de l'université Paris Saclay est-elle aussi cofondatrice et directrice scientifique de Quandela. De même, Antoine Browayes de l'Institut d'optique est lui directeur de recherche au CNRS et cofondateur de Pasqal.

En plus de Bpifrance et du fonds d'investissement Quantonation, le *Deep Tech Founders* forme des entrepreneurs et chercheurs dans les *deep techs*. C'est un programme international créé par l'équipe de Hello Tomorrow. L'écosystème quantique, côté entreprises, est fédéré par Le Lab Quantique, créé conjointement par Quantonation et Bpifrance. Il organise des rencontres régulières qui réunissent principalement des entrepreneurs des technologies quantiques, français et étrangers. Ces rencontres ont eu lieu sous forme de visioconférences sur Zoom pendant la période de la pandémie de covid-19 en 2020 et 2021.

L'une des spécificités de la France en Europe est qu'on y trouve de grandes entreprises qui ont directement investi dans les technologies quantiques et les technologies habilitantes quantiques (cf. figure 9) : Atos (logiciels, émulateurs, accélérateurs quantiques), Thales (capteurs), Air Liquide (cryogénie), Orano (isotopes), Radiall



FIGURE 9. Grandes entreprises françaises ayant investi dans les technologies quantiques.

(câblage) et un bon nombre dans la photonique (comme ixBlue, Azurlight, Aurea et Cailabs) et même les machines de production de semi-conducteurs avec Plassys Bestek et Riber.

Bien sûr, la France accueille également Atos qui s'est lancé dans l'informatique quantique en 2016 sous la direction de Thierry Breton en créant des outils d'émulation dont un serveur aQML et des outils de développement. Ils prévoient de devenir des intégrateurs d'accélérateurs quantiques et de supercalculateurs classiques pour construire des solutions hybrides.

Dans chaque pays et à l'échelle internationale, l'enjeu clé est dans un premier temps de favoriser les collaborations interdisciplinaires. Les technologies à créer et assembler pour créer un ordinateur quantique sont très nombreuses aussi bien du côté physique et matériel que du côté des outils de développement et du logiciel. Le tout avec une incertitude scientifique qui subsiste sur la capacité de la recherche mondiale à créer des ordinateurs quantiques qui montent en échelle et dépassent réellement les capacités des ordinateurs classiques.

L'enjeu suivant rejoint ceux des autres technologies du numérique et relève plus de la dimension business et marketing. Elle comprend notamment la création d'écosystèmes logiciels et de ce point de vue-là, les USA ont une certaine avance.

Bataille des compétences

Enfin, on ne peut pas parler d'écosystème quantique sans évoquer l'enjeu des compétences. La complexité du domaine est élevée. Sa compréhension n'est pas donnée à tout le monde, y compris chez les ingénieurs. Un bon bagage scientifique est indispensable pour appréhender les concepts de base de la physique quantique et

ceux de l'information quantique. Il faut se remettre aux mathématiques, notamment à l'algèbre linéaire et au calcul matriciel. Les concepts sont souvent contre-intuitifs et plein de faux amis. Les modèles de programmation sont complètement différents du développement logiciel sur ordinateur classique.

Les talents à même de s'appropriier le sujet sont rares et ils ne peuvent pas être formés rapidement comme on a pu le faire sur le développement web avec des initiatives telles que Simplon.co ou l'école 42. La pluridisciplinarité et le croisement des compétences seront aussi indispensables. De nombreux pays, dont la France, ont ainsi lancé de nouveaux cursus de formation sur les technologies quantiques et leurs applications. Ils vont s'arracher les talents à la fois du côté de la recherche, qui est mieux financée, que de celui des startups, des grandes entreprises technologiques puis des entreprises utilisatrices.

On peut saluer en France les initiatives lancées à différents endroits : à l'université Paris-Saclay avec son master quantique ARTEQ, à l'université Grenoble Alpes (UGA), dans plein d'universités en région (Strasbourg, Bordeaux, Nancy, Lyon, Nice, ...) ainsi que dans de nombreuses grandes écoles telles que Polytechnique, Télécom Paristech, CentraleSupélec, les Écoles normales supérieures (impliquées depuis longtemps dans la filière), l'INSA Lyon ainsi que l'EPITA.

À chaque nouvelle vague technologique, sa vitesse d'appropriation par les entreprises du côté de l'offre comme de la demande est en effet clé pour en tirer parti de manière compétitive.



Fondation Blaise Pascal¹

Objectifs. — *Donner à aimer l'informatique et les mathématiques, voilà le défi relevé par la fondation Blaise Pascal (FBP), sous égide de la fondation pour l'université de Lyon (FPUL), depuis 2016. La FBP est une fondation à vocation nationale, qui a pour objectif de promouvoir, soutenir, développer et pérenniser les actions de médiation scientifique en informatique et mathématiques à destination de toute citoyenne et de tout citoyen, et plus spécifiquement auprès des jeunes. La fondation porte quatre missions :*

- *agir en faveur de l'inclusion par les sciences ;*
- *former à la démarche scientifique dès l'école, et en dehors de l'école ;*
- *démultiplier l'impact sur l'ensemble du territoire des actions de médiation ;*
- *sensibiliser au numérique responsable et éthique.*

Avec un budget annuel de 250 k€, la FBP développe son impact partout sur le territoire. Depuis sa création, elle a soutenu 258 projets pour plus de 1 million d'euros alloués au profit de plus de 2 millions de personnes.

La médiation, un enjeu d'avenir. — De la recherche fondamentale aux moindres actions de notre quotidien, l'informatique et les mathématiques sont présentes partout. Le monde actuel fonctionne et se développe à l'intersection des mathématiques et de l'informatique et l'échange entre ces deux disciplines. Avec la transition numérique de notre société, et les nouveaux défis survenus récemment avec la crise sanitaire liée à la COVID-19, ces compétences scientifiques sont, aujourd'hui et plus que jamais, des compétences clés. Alors que la France est reconnue pour son excellence scientifique, les filières en mathématique et informatique de l'enseignement supérieur rencontrent un succès limité. Un constat qui devient un enjeu de société

1. <https://www.fondation-blaise-pascal.org>.



FIGURE 1. Clubs de programmation pour jeunes organisés par l'Exploradôme, le musée des sciences et du numérique à Vitry-sur-Seine (94). ©Exploradôme

majeur, tant les débouchés de ces disciplines vont croissant et les candidats se font rares. De plus, un déficit d'image auprès des jeunes femmes est largement constaté ainsi qu'une pénurie de main d'œuvre féminine qualifiée dans ces domaines. La situation est problématique et l'évolution numérique ne peut pas passer à côté de la moitié de la population.

Une fondation en mouvement. — Le projet Cap' Maths, financé dans le cadre du programme « Investissements d'avenir » entre 2012 et 2016, a largement contribué à améliorer la coordination et la visibilité des actions existantes et à en susciter de nouvelles. Fondée initialement par le CNRS et l'université de Lyon, rejoints en 2019 par Inria, la fondation Blaise Pascal est née de cette volonté de pérenniser et de continuer à développer les actions portées par ce projet. Son président, Serge Abiteboul, pilote le conseil stratégique formé de représentants des fondateurs et de la société civile. Ce conseil coordonne la politique de la FBP et valide les décisions du conseil scientifique. Ce dernier est composé d'experts en médiation en informatique et mathématiques qui représentent, notamment, les sociétés savantes de ces deux disciplines, ainsi que l'enseignement secondaire. Il est chargé de veiller à la qualité scientifique des projets de médiation soutenus par la fondation. Pour mener à bien les missions qui lui ont été fixées, la FBP met en œuvre :

- la lutte contre les préjugés et les stéréotypes sociaux et de genre qui empêchent certains jeunes de se lancer dans des études en informatique et en mathématiques ;
- la structuration des actions de diffusion dans ces deux disciplines et la mutualisation des moyens en s'appuyant sur tous les acteurs de la médiation scientifique ;

- la recherche de mécènes et la levée de fonds pour soutenir financièrement les actions de médiation ;
- l'évaluation de l'impact de ces actions, en particulier en termes d'égalité des chances ;
- la réalisation d'un maillage du territoire national afin d'atténuer les disparités géographiques et sociales dans l'accès à la connaissance scientifique ;
- l'éveil aux vocations de médiation scientifique chez les chercheurs et enseignants-chercheurs par la formation mais aussi par la valorisation de celle-ci dans leur carrière.

Les appels à projets. — Afin de soutenir la médiation en informatique et mathématiques sur l'ensemble du territoire, d'inciter les régions de France les plus isolées à initier des actions et garantir une certaine équité entre les acteurs, le FBP lance deux appels à projets par an, l'un générique et l'autre dédié aux écoles d'été, stages et clubs pour des jeunes de collèges ou de lycée.

Ainsi, de sa création à 2021, grâce au soutien financier de ses fondateurs et au conventionnement avec l'ANR, la FBP a utilisé le reliquat du PIA Cap' Maths, et a lancé dix appels à projets. La FBP a soutenu 147 structures et a alloué près de 1 million d'euros aux acteurs de médiation via ces appels et a redistribué plus de 120 000 € de fonds fléchés émanant directement des entreprises. Au total, 258 projets innovants et pluriannuels ont été subventionnés sur l'ensemble du territoire national : des stages, des écoles d'été, des clubs, des festivals, des pièces de théâtre, des expositions, des vidéos, etc.

Grâce à ses appels à projets et son travail de maillage du territoire, la FBP a identifié plus de 300 acteurs de médiation sur toute la France.

Depuis 2017, la FBP a réussi à récolter plus de 355 000 € de fonds privés à destination de la médiation en informatique et mathématique. Les principales entreprises qui ont collaboré avec la FBP sont Alstom, Sopra Steria, APL, Automation Anywhere, la fondation Michelin, la fondation Tezos, Google for Education, Keyrus, SNCF. Par ailleurs, la fondation compte également des partenaires institutionnels tels que Télécom ParisTech, le pôle diversité et réussite de l'École polytechnique, et des partenaires associatifs tels que Talents du numérique, Class Code, Animath, Femmes & mathématiques.

En s'associant à des acteurs avec qui elle partage la même ambition, la fondation œuvre et développe ses propres projets. Ainsi, la fondation est aussi lauréate de nombreux appels à projets externes comme ceux lancés par la fondation Femmes@Numérique ou la fondation SNCF.

Les quatre grands projets menés actuellement par la FBP sont :

- « 1 scientifique – 1 classe : Chiche ! » en partenariat avec Inria, la fondation Inria, le CNRS et le ministère de l'Éducation nationale, de la Jeunesse et des Sports et soutenu par Alstom et Roadef. Par cette opération d'envergure



FIGURE 2. Opération « Filles, maths et informatique : une équation lumineuse », une initiative d'Animath et Femmes & Mathématiques, soutenue par la fondation Blaise Pascal. ©Femmes & Mathématiques

nationale, elle veut encourager le goût des élèves de seconde pour les sciences du numérique grâce à la rencontre avec des scientifiques illustrant de manière pédagogique et attractive « la recherche » dans ce domaine et son impact pour la société ;

- journées « Filles, maths et informatique : une équation lumineuse » en partenariat avec Animath et Femmes&mathématiques. Il s'agit pour ces journées déjà renommées de les faire changer d'échelle (cf. figure 2). Dédiées uniquement aux filles, elles permettent par de multiples activités de déconstruire les stéréotypes sociaux de genre liés aux métiers du numérique et de promouvoir les carrières dans ces domaines auprès des filles ;
- depuis 2020, la fondation Blaise Pascal organise des écoles d'été pour lycéens dont la première édition se tiendra en juillet en Haute-Loire et en Ardèche sur les thèmes mathématiques et astronomie en partenariat avec le pôle diversité et réussite de Polytechnique ;
- en 2018, la FBP initie une collaboration avec l'entreprise Sopra Steria en fondant, notamment, le cercle Féminisons les mathématiques et l'informatique (FMI). Ce groupe a pour objectif d'analyser la situation des femmes dans le secteur des mathématiques et de l'informatique afin de proposer des actions pertinentes de sensibilisation des filles à ces métiers. Quelques chiffres évocateurs : en 2016, seulement 12 % des professeurs d'université en mathématiques étaient des femmes, 19 % en informatique et seules 9 % des *startups* françaises étaient dirigées par des femmes.



Maison des mathématiques et de l'informatique²

Objectifs. — *Créée en 2012 par le Laboratoire d'excellence en mathématiques et informatique fondamentale de Lyon (Labex MILYON) porté par l'université de Lyon dans le cadre de l'IDEXLYON, la Maison des mathématiques et de l'informatique utilise des expositions, contes, ateliers, jeux, spectacles, conférences, etc. pour permettre à tous et toutes de découvrir et comprendre le monde qui nous entoure.*

Pionnière du genre, la MMI est un centre de médiation des savoirs dédié aux sciences mathématiques et informatiques via une approche vivante, ludique et pluridisciplinaire. Entièrement piloté par des enseignants-chercheurs passionnés, c'est un lieu où convergent : science, art, musique, histoire, architecture... pour une expérience nouvelle des mathématiques et de l'informatique !

Retrouvez toutes les informations sur notre site web.

Nouvelle exposition. — Vous avez dit IA ? Mais qu'est-ce que l'intelligence artificielle ? Que recouvre ce concept qui fait tant parler de lui ? Quand est-il apparu, comment s'est-il développé ? Comment fonctionnent les IA actuelles et quels sont leurs domaines de prédilection ? C'est à ces multiples questions que l'exposition « Entrez dans le monde de l'IA » entend apporter des éléments de réponse. De son histoire à la notion elle-même, de sa théorisation à ses applications, la diversité de l'IA est révélée ainsi que les enjeux qu'elle représente pour l'avenir. Rendez-vous jusqu'au 25 juin 2022 pour découvrir cette nouvelle exposition !

2. <https://mmi-lyon.fr>, Facebook : @mmi.lyon, Twitter : @MMI_lyon, Instagram : mmi.lyon.



GDR Informatique géométrique et graphique, réalité virtuelle et visualisation (IG-RV)¹

Objectifs. — *Le domaine de recherche couvert par le GDR Informatique géométrique et graphique, réalité virtuelle et visualisation se situe majoritairement en science informatique. Les objets « 3D », ou plus généralement les mondes virtuels, sont les enjeux de la plupart des études du GDR IG-RV. Ces études visent à capturer ou créer les formes géométriques, leurs apparences ou leurs mouvements, pour ensuite offrir des moyens d’interagir avec ces mondes. Cette tâche se caractérise par son extrême complexité, que ce soit pour capturer des éléments du monde réel ou pour créer des mondes virtuels complexes.*

Pour créer des mondes virtuels, chaque élément est représenté par des données numériques qui seront traitées, modifiées, analysées ou synthétisées, mais aussi visualisées ou animées, et avec lesquelles l’interaction devra être opérée en temps réel. La capture d’éléments du monde réel participe à sa modélisation numérique en s’appuyant sur le traitement d’informations perçues ou captées sur l’utilisateur.

L’objectif est de se rapprocher toujours plus du monde réel ou de l’augmenter d’éléments numériques ou captés permettant de placer l’humain au cœur des systèmes, en lui proposant des retours sensoriels et toujours plus de contrôle, par exemple des phénomènes visualisés ou des avatars d’une réalité virtuelle.

1. <https://gdr-igrv.icube.unistra.fr/>.

Présentation et Actualités

Le GDR IG-RV en quelques chiffres. — Le GDR IG-RV représente environ 750 personnes (44 unités dont 29 UMR, 68 équipes, 425 permanents, 327 doctorants) et est structuré en 6 groupes de travail (GT) :

- Géométrie discrète et morphologie mathématique (GT GDMM);
- Modélisation géométrique (GT GM);
- Animation et simulation (GT AS);
- Rendu (GT Rendu);
- Analyse visuelle de données (GT Visu);
- Immersion, interaction et usages en RV/RA (GT Réalités virtuelles).

Plénière IGRV2021. — Les Quatrièmes journées plénières du GDR IG-RV ont eu lieu à distance la semaine du 16 au 19 mars 2021 concomitamment à plusieurs journées de GT. L'appel à volontaires est d'ores et déjà lancé pour l'organisation des journées plénières biennuelles de 2023.

Prix de thèse IGRV. — Depuis 2017, le GDR IG-RV a décidé de mettre en place un prix de thèse IGRV avec le soutien des associations AFIG, AFRV et EGFR. L'objectif de ce prix de thèse est de récompenser chaque année une excellente thèse issue de la communauté du GDR IG-RV. L'intérêt est double. Pour le jeune docteur concerné ou la jeune docteure concernée, ce prix apporte une grande visibilité à ses travaux et une reconnaissance de la communauté du GDR sur l'excellence des résultats obtenus. Pour la communauté du GDR, ce prix permet de rendre visible très largement des travaux de premier ordre issus de nos thématiques de recherche. Le contour disciplinaire du prix est celui du GDR et de ses groupes de travail. Ce prix de thèse du GDR IG-RV est parrainé par la SIF qui encourage les candidats de nos domaines à déposer des dossiers au prix de thèse Gilles Kahn, et en particulier le lauréat ou la lauréate et les éventuels accessits du prix de thèse du GDR IG-RV.

Historique et évolution du GDR IG-RV. — Le GDR IG-RV a 15 ans. Créé en 2006 sous la direction de Jean-Pierre Jessel avec la seule communauté de l'informatique graphique, son élargissement à la réalité virtuelle (mais pas seulement, également à la visualisation scientifique et aussi, à la géométrie discrète) a été porté pendant 8 années de 2014 à 2021 par Dominique Bechmann, assurant ainsi un continuum scientifique qui a du sens et dont la communauté n'a pas fini de tirer les bénéfices. En janvier 2022, c'est un groupement de recherche uni autour de l'informatique géométrique et graphique, la réalité virtuelle et la visualisation qui devrait prendre une nouvelle dimension, sous la direction de David Coeurjolly, en animant et en conduisant la prospective scientifique de nos domaines.



La numérisation de l'école, fausse bonne idée du siècle

Philippe Bihoux¹

Notre société fortement numérisée, vit des mutations profondes, économiques, sociales, culturelles, et l'école se doit de préparer au mieux les futurs citoyens au monde de demain. Mais généraliser l'enseignement sur écran, équiper chaque élève d'un objet informatique, dès le plus jeune âge, sont-ils les meilleurs moyens de « préparer les jeunes à devenir des individus autonomes, créatifs et responsables », comme s'est donné pour mission le ministère de l'Éducation nationale ? Il serait temps d'ouvrir réellement le débat.

Philippe BIHOUX Karine MAUVILLY

LE DÉSASTRE DE L'ÉCOLE NUMÉRIQUE

Plaidoyer pour une école sans écrans



POINTS

Un peu d'histoire... : de la genèse de l'école numérique

L'expression « école numérique » est entrée dans le langage courant et les rapports officiels au début de la décennie 2010 (auparavant, on parlait plutôt d'informatique à l'école...), notamment à partir du « grand plan numérique pour l'école de la République », annoncé par le Président de la République, François Hollande, en septembre 2014, un plan qui devait « contribuer à rendre [l'école] plus efficace, plus juste et plus inclusive »². Il consistait à équiper, en quelques années, l'ensemble des 3,3 millions

1. Ingénieur et essayiste, directeur général du groupe Agence d'architecture interdisciplinaire (www.arep.fr), co-auteur, avec Karine Mauvilly, de *Le désastre de l'école numérique. Plaidoyer pour une école sans écrans* (Seuil, 2016; Points 2021).

2. Discours de M^{me} Najat Vallaud-Belkacem, ministre de l'Éducation nationale, 7 mai 2015.

de collégiens d'une tablette ou d'un ordinateur portable, et à adapter les méthodes et pratiques d'enseignement — cours, exercices, contrôle des connaissances... — dans toutes les disciplines, pour utiliser le vecteur numérique.

Comme la plupart des « grands plans » gouvernementaux, la mise en œuvre n'a suivi que partiellement les effets d'annonce. Parallèlement à l'augmentation du taux d'équipement des élèves et à l'évolution indéniable des pratiques pédagogiques (utilisation accrue de la vidéo en classe, expérimentations de classe inversée³, travaux à la maison et rendus sur ordinateur...), la vie à l'école s'est également numérisée, avec les logiciels d'appel, le cahier de texte et la saisie de notes électroniques, les « espaces numériques de travail » (ENT) où les professeurs déposent des documents et communiquent par messagerie avec les familles.

En termes d'innovation — et de fascination — « techno-pédagogique », l'école n'en est pas à son coup d'essai. Depuis 150 ans, à chaque fois qu'un nouveau médium a été mis au point, elle s'est emballée pour les formidables opportunités qui s'ouvriraient, séduite, souvent, par les fournisseurs de matériel lui promettant monts et merveilles. Il est d'ailleurs frappant de constater à quel point les mêmes arguments, à travers les âges, ont été brandis... jusqu'à aujourd'hui même.

L'histoire démarre à la fin des années 1870 avec Alfred Molteni, un fabricant de lanternes magiques (ancêtres de PowerPoint), qui vante les images animées à vertu éducative, publie un *guide pratique spécialement destiné aux instituteurs* et organise des conférences devant les directeurs d'écoles normales et les inspecteurs de l'enseignement primaire. Il y vante la facilitation de l'enseignement, l'instruction en s'amusant, l'ouverture d'esprit et la motivation suscitées chez les élèves. Quelques milliers de lanternes seront commandées et quelques tonnes de diapositives fabriquées, mais déjà, une autre technique menace de chasser cette première innovation, au tout début du XX^e siècle : le film cinématographique. Thomas Edison, le fameux inventeur et industriel, fabricant d'objets électriques en tout genre, et entre autres des projecteurs de cinéma, pronostique en 1913 que « bientôt les livres seront obsolètes dans les écoles » et promet même, quelques années plus tard, d'obtenir « 100 % d'efficacité » dans l'éducation du futur — que cela signifie-t-il ? mystère... — grâce à l'image animée.

Le cinéma à l'école n'aura pas le résultat escompté, mais heureusement, la radio prend le relais au début des années 1930. Il s'agit cette fois de permettre à tous de profiter des « services des meilleurs professeurs [et de] l'inspiration des plus grands leaders ». Un peu comme les MOOC⁴ de nos universités les plus modernes ! À partir des années 1950—1960, c'est la télévision qui reprend le flambeau, mais là encore,

3. Cours préparés par un visionnage de vidéos à la maison, afin de consacrer le temps de classe à des exercices ou des approfondissements.

4. Massive Online Open Courses.

les prophéties les plus enthousiastes sur « l'éducation audiovisuelle » seront démenties par des expériences décevantes. On peut également mentionner, dans les années 1960, les « machines à enseigner », des dispositifs électromécaniques développés par le psychologue behavioriste Burrhus F. Skinner. Celles-ci allaient permettre l'individualisation des apprentissages, un serpent de mer qui va revenir souvent ensuite ; mais elles tombent dans l'oubli avec l'arrivée de la mini puis la micro-informatique.

A partir des années 1970, les « plans » informatiques s'enchaînent : 58 lycées et collèges pilotes sont d'abord équipés, entre 1973 et 1976, de mini-ordinateurs, des Mitral 15 de CII et des T1600 de Télémécanique (200 kg et 2 mètres de hauteur, tout de même) ; en 1979 est lancée l'opération « 10 000 micro-ordinateurs » ; en 1983 on passe au plan « 100 000 ordinateurs et 100 000 enseignants formés », qui évolue en « plan informatique pour tous », basé sur les TO7 de Thomson (et leurs mythiques crayon optique et clavier récalcitrants). Le plan est abandonné l'année suivante, avec le changement de gouvernement, et l'informatique à l'école connaît une petite traversée du désert... avant de revenir en force à la fin des années 1990, à travers de multiples annonces, plans, rapports officiels, soutiens à l'équipement des établissements aux différentes échelles territoriales, qu'il serait trop long d'énumérer ici.

Pendant tout ce temps, bien sûr, les discours pédagogiques fleurissent, sur la nécessité de s'ouvrir sur le monde, sur l'école qui a toujours un métré de retard sur la vie et doit « s'adapter », sur les incroyables opportunités de l'informatisation de l'enseignement. Un exemple parmi mille, exhumé d'un des rares ouvrages critiques de l'informatisation de l'école dans les années 1980, *Arsenic et jeunes cervelles*, de la part du journaliste et homme politique Jean-Jacques Servan-Schreiber, grand promoteur de l'informatisation du pays à la fin des années 1970 (quels cancre nous étions alors, déjà en retard sur les États-Unis !) : « *L'informatisation va permettre de réaliser la prédiction de Jaurès [...] La société informatisée, ce sera le jaillissement de la vie ardente et libre de l'humanité qui s'appropriera l'univers par la science, l'action et le rêve* »⁵. Quarante ans plus tard, dans cette société informatisée — pardon, numérisée... — advenue ou peu s'en faut, le rêve laisse quand même songeur...

Quels enseignements tirer de cette aventure ? Chaque nouvelle technologie devait « révolutionner » l'éducation. La première question est évidemment celle de l'efficacité pédagogique, la plus légitime, celle qui justifie tout ce battage, tout cet argent dépensé en matériels — rapidement frappés d'obsolescence évidemment, quand ils ne restent pas stockés dans des placards après quelques essais — : le numérique permet-il, avec les moyens actuels (car pour les plans passés, la messe est dite...), de mieux apprendre ?

5. Jean-Pierre Despins et Marie-Claude Bartholy, Union Générale d'édition, 10/18, 1987, p159.

L'épineuse question pédagogique

Les promoteurs du numérique à l'école font de nombreuses promesses : motivation et concentration accrues, amélioration des performances, possibilités de travail collectif, pédagogies actives ou ludiques, adaptation au rythme de chaque enfant, ressources pédagogiques enrichies... Pour les élèves dissipés, le numérique permettrait d'augmenter la motivation, l'envie d'apprendre, la concentration. Il transformerait les élèves en premiers de la classe, bosseurs, curieux, collaboratifs. Systématiquement associé à une pédagogie interactive et ludique, le numérique permettrait aussi de redonner confiance aux décrocheurs. Et en fournissant des ressources pédagogiques « illimitées », il permettrait même de lutter contre les inégalités.

Mais ces prétendus avantages pédagogiques du numérique ne sont, aujourd'hui comme hier, que des promesses, voire des espoirs : face à la « crise » (sans fin) de l'école, on n'a plus grand-chose d'autre à proposer, le numérique devient le dernier cache-misère des réformes (et de leurs échecs) qui s'enchaînent dans l'Éducation nationale. Malgré les effets d'annonce, les miracles du numérique n'ont été corroborés par *aucune* étude scientifique — aucune ! Et pour cause ; on imagine difficilement comment on pourrait prendre deux groupes témoins d'élèves, suffisamment grands pour être statistiquement représentatifs et corrigés d'éventuels biais contextuels et sociaux, et les soumettre, pendant une dizaine d'années de scolarité, à une expérience grandeur nature, l'un à une éducation non numérique et à des dictées sur cahier à grands carreaux, l'autre à une dose massive d'écrans et à des *twitées* collaboratives, pour comparer ensuite leur niveau d'orthographe de jeune adulte...

Il faut donc s'en tenir à deux types d'analyses. La première consiste à examiner l'évolution générale du niveau scolaire dans les différents pays, en essayant de le corréler à l'effort de numérisation des systèmes scolaires. C'est ce qu'a tenté de faire l'OCDE à partir de 2015 (sur la base des résultats de l'enquête OCDE / Pisa 2012). Patatras : « *En moyenne, au cours des dix dernières années, les pays qui ont consenti d'importants investissements dans les TICE⁶ n'ont enregistré aucune amélioration notable des résultats de leurs élèves [...]* ». Au contraire, « *les niveaux d'utilisation supérieurs à la moyenne des pays de l'OCDE sont associés à des résultats significativement plus faibles* »⁷. Autrement dit, les élèves utilisant très souvent les ordinateurs à l'école obtiennent des résultats inférieurs à la moyenne : plus les élèves travaillent sur écran, moins ils comprennent ce qui est écrit dessus. Évidemment, il faut prendre des précautions dans l'interprétation de ces chiffres : on manque encore de recul, on n'est qu'au début de l'aventure, les professeurs n'ont pas encore pris toute la mesure des possibilités pédagogiques du numérique, etc. Les arguments ne manquent donc pas pour expliquer ce pitoyable résultat, à date.

6. Technologies de l'information et de la communication appliquées à l'enseignement.

7. OCDE/Pisa, *Connectés pour apprendre ? Les élèves et les nouvelles technologies*, 2015.

Les secondes analyses consistent en des études scientifiques plus ponctuelles, qui ne plaident pas vraiment en faveur du numérique ; les résultats sont en tout cas partagés. Plusieurs recherches montrent par exemple qu'il ne faut pas confondre motivation pour un apprentissage et motivation pour le support ; que les supports trop riches, ou animés, sont parfois trop exigeants pour le cerveau par rapport à de simples schémas ; que la prise de notes à la main permet une meilleure mémorisation que la dactylographie ; que l'attention se détourne du fond vers le fonctionnement de l'outil. Si le numérique a une certaine capacité de sidération pour les élèves, ça ne signifie pas qu'il favorise l'apprentissage.

On peut enfin tenter de répondre pied à pied aux arguments développés par les tenants du numérique. Oui, il y a une crise de l'attention et de la concentration à l'école, qui pourrait le nier ? Les outils numériques peuvent, avec l'effet de surprise, créer un certain enthousiasme, au début du moins... mais la crise de la concentration n'a-t-elle pas été en partie créée par la multiplicité des écrans en dehors de l'école, et cette « société du multi-tâches », qu'on importe maintenant à l'école ? Qui peut croire en effet qu'en proposant aux élèves de sortir et d'utiliser leur smartphone en classe — on appelle ça BYOD⁸ —, ils n'en profiteront pas pour papillonner ailleurs et regarder aussi les dernières notifications tombées sur les réseaux sociaux ?

Certes, la dimension du jeu est importante, surtout chez les petits. Mais une pédagogie peut être ludique sans le numérique. Sous prétexte du manque de motivation des élèves, il faudrait tout « gamifier », adapter les exercices pour éviter de les démotiver. On perd la dimension de l'effort, qui avait quand même un potentiel intérêt pédagogique : si tout est toujours divertissant, prémâché, quelle sorte de citoyens formons-nous ? Le monde n'est pas tout rose. À partir d'un certain âge, face à un enseignement parfois ennuyeux ou difficile, l'élève peut aussi développer son sens critique, apprendre la patience, trouver du plaisir dans la réussite (cela ne veut pas dire évidemment que l'école doit toujours être ennuyeuse)...

En 1933, l'écrivain Georges Duhamel s'insurgeait déjà contre une école du moindre effort qui deviendrait possible grâce à la radio et au cinéma, une éducation « sans douleur » : « *Dès qu'il s'agit d'éducation et d'instruction, l'effort compte seul. Il est des sujets qui ne peuvent apprendre qu'au prix d'un grand effort, il en est qui font merveille au prix d'un effort minime. Les uns ont l'effort joyeux, d'autres l'effort triste. N'importe ! Il faut un effort. C'est par l'effort que l'esprit s'exerce et s'améliore. C'est la discipline de l'effort qui permet à l'homme de viser sans cesse plus haut, d'affronter sans cesse des épreuves plus rudes. Les connaissances acquises par quelque artifice plaisant, sans peine et sans gymnastique, sont instables, labiles, de faible usage, de petit profit. L'homme justement ambitieux souhaite*

8. Bring your own device.

sans doute le succès, mais, d'abord, d'avoir mérité le succès »⁹. Aux pédagogues d'aujourd'hui, tout cela paraîtra sans doute un peu vieillot, évidemment.

Quant au numérique qui permettrait de lutter contre les inégalités, c'est à voir. Tout le monde s'accorde sur le fait que la « fracture numérique » ne réside plus dans l'équipement, ou dans l'accès au réseau à haut débit, mais dans les usages et l'accompagnement parental. Un argument — valable — pour *former au numérique*. Mais le ministère ambitionne plutôt de *former par le numérique*. Or la numérisation du travail à la maison exige un suivi parental plus appliqué. Par exemple, tous les élèves ne visionnent pas une vidéo de la même façon : certains sont concentrés, peut-être accompagnés par leurs parents, d'autres regardent d'un œil, en surfant sur les réseaux sociaux ; tous les élèves ne se mettent pas sur l'ordinateur pour effectuer des recherches avec la même application, ou la bonne méthode... Toute demande de travail sur écran à la maison est donc susceptible de creuser les inégalités. Les enfants les moins favorisés sont aujourd'hui les plus connectés, du fait d'un certain retrait éducatif des parents au profit des écrans.

Enfin, le numérique n'a pas le monopole de l'innovation pédagogique : on peut produire du contenu sans *twictée*, apprendre de façon ludique sans *serious games* électroniques, faire de la « classe inversée » avec une lecture documentaire plutôt qu'une vidéo. Quant à la richesse des ressources pédagogiques, la « pauvreté » des manuels en papier a-t-elle déjà été pointée comme l'une des causes de la crise de l'école ?

Alors pourquoi river les enfants à des machines, dès le plus jeune âge, « pour quelles raisons, au juste, faut-il à tout prix remplacer livres et cahiers — bientôt les enseignants — par des écrans ? », questionne la tribune initiée, fin 2020, par *Lève les yeux* et *Nous personne* et signée par une vingtaine d'associations, après avoir envoyé deux courriers au gouvernement, aux présidents d'exécutifs locaux, et à l'ensemble des parlementaires : « [...] un nombre chaque jour croissant de parents, d'enseignants et autres personnels de l'Éducation nationale découvrent avec effroi les effets néfastes de la surexposition aux écrans et refusent que l'école républicaine se transforme, elle aussi, en kaléidoscope géant, après nos maisons, nos gares et nos rues »¹⁰.

En 1983, Maurice Nivat, professeur d'informatique à Paris VII, chargé d'une mission « Informatique fondamentale et programmation », porta un regard pour le moins mesuré sur l'EAO (enseignement assisté par ordinateur), comme on disait à l'époque.

9. Georges Duhamel, *L'humaniste et l'automate*, Paul Hartmann éditeur, 1933, p186.

10. Tribune collective. « La numérisation de l'école est nuisible aux enfants et à l'environnement », *Reporterre.net*, 12/12/2020.

Il déclara dans une interview à Libération que celui-ci « *ne diffuse pas d'autre savoir que de taper sur un clavier et de lire une réponse sur un écran* »¹¹. On en est toujours un peu au même point. Quant aux jeux pédagogiques, le rapport n'était pas tendre avec eux : « *[Leur] vertu pédagogique est nulle ou plutôt exactement égale à celle des machines à sous, "flippers", "zinzins" que contiennent tous les cafés de France et de Navarre (et dont personne n'a jamais songé à faire un des éléments de la formation des jeunes, encore qu'ils exigent de l'adresse et des réflexes, deux choses dont les élèves ont beaucoup de chance d'être mieux pourvus que leur professeur, ce qui explique sans doute qu'ils y vont rarement jouer ensemble et sûrement pas dans le cadre d'un programme pédagogique)* »¹². Depuis, les rapports officiels ont perdu en fraîcheur et en honnêteté...

Effets délétères

L'école numérique soulève par ailleurs d'énormes questions sanitaires, écologiques et sociales. Des questions sanitaires, d'abord, car les preuves des « effets délétères » de la surexposition aux écrans sur la jeunesse s'accumulent. Les enfants et les adolescents passent déjà beaucoup — trop — de temps devant des écrans. Dans ce contexte, que fait l'école numérique ? D'une part, elle augmente le temps global d'écran des enfants, d'autre part, en demandant aux élèves de se connecter après l'école pour faire leurs devoirs, elle *légitime* auprès des parents l'usage des écrans, elle *engendre* un besoin d'équipement supplémentaire (ordinateur, imprimante...).

Or des études nombreuses (plus de 1500 études internationales) pointent les risques psychosociaux d'une surexposition aux écrans, en particulier pour les plus jeunes (alors que la numérisation gagne désormais l'école primaire, et que certains expérimentent dès la maternelle) : addiction, dépression, agitation, difficultés de concentration, troubles de l'attention, troubles cognitifs, intolérance à la frustration, baisse de l'empathie, violence... La myopie gagne les pays riches du fait d'un manque d'exposition à la lumière naturelle ; le temps de sommeil des jeunes est globalement réduit ; les écrans, nécessaires dorénavant pour faire ses devoirs, ne quittent plus les chambres et exposent les ados et même les primaires à des images inappropriées. Un enfant harcelé à l'école est désormais un enfant harcelé dans sa chambre, par écran interposé.

L'Agence de sécurité sanitaire française (Anses), qui classe les électro-fréquences « cancérogènes possibles pour l'homme » tandis qu'on généralise le Wifi dans les classes, mentionnait même, dans un rapport de 2016¹³, les effets possibles de l'usage

11. Interview du 22/11/1983. Cité dans Cédric Biagini, Christophe Cailleaux, François Jarrige, *Critiques de l'école numérique*, L'échappée, 2019, p.390.

12. Maurice Nivat, *Mission informatique et programmation*, La documentation française, 1983, p.43.

13. Anses, *Exposition aux radiofréquences et santé des enfants*, juillet 2016.

des téléphones et tablettes sur le bien-être et les fonctions cognitives (mémoire, fonctions exécutives, attention). Plusieurs ouvrages sont également parus ces dernières années, sur le danger des écrans dans la petite enfance ou sur les risques sanitaires liés à l'addiction numérique¹⁴.

Des questions écologiques, ensuite, car l'empreinte du numérique est forte, loin de l'illusion d'immatérialité. Avec ses infrastructures, serveurs, antennes-relais, bornes Wifi, routeurs, câbles terrestres et sous-marins, centres de données, et tous les équipements des utilisateurs, souvent remplacés, le numérique consomme plus de 10 % de l'électricité mondiale, des métaux comme l'argent, l'étain, le cobalt, le tantale ou les terres rares, génère des déchets électroniques très difficiles à recycler, qui finissent pour l'essentiel en décharge ou en incinérateur — seuls 15 % des déchets électriques et électroniques sont recyclés dans le monde —, ou dans les bidonvilles du Ghana et de Chine, où les polluants se déversent dans les sols et les rivières.

Évidemment, ce coût environnemental est lié à l'ensemble du numérique. Il ne viendrait à personne l'idée de contester les avancées technologiques dans des domaines comme la médecine, par exemple, de dénoncer les appareils électroniques chez les dentistes, les radiologues ou dans les hôpitaux, au prétexte qu'on opérerait mieux les patients et que les dents étaient mieux soignées *avant* ! Mais dans le domaine éducatif, nous sommes bien loin d'avoir démontré une quelconque utilité à la course en avant technologique. Outiller tous les élèves et déployer l'infrastructure nécessaire, c'est gâcher des ressources précieuses, impacter un peu plus l'environnement, pour un résultat pédagogique *probablement moins bon*, alors même que « l'éducation au développement durable » fait désormais partie du socle commun de connaissances...

Des questions sociales, enfin, car les milliards d'euros injectés dans le numérique, passant en grande partie dans les efforts d'équipement et les licences de logiciels, partent dans une production surtout extraterritoriale. De façon alternative à cette coûteuse course en avant technologique – perdue d'avance car l'obsolescence technique des outils digitaux est rapide –, des projets davantage créateurs d'emploi pourraient pourtant être imaginés : dédoublement de certaines classes, accès gratuit à des activités artistiques (musique, théâtre, dessin, sculpture...), où résident aussi les inégalités, etc.

Mais l'ambiance est plutôt à la réduction du nombre de postes : à moyen terme, les rapports très officiels mentionnent déjà la possibilité d'une évolution vers un « mode mixte », avec une partie des cours en « présentiel » (des vrais profs dans les

14. Manfred Spitzer, *Les ravages des écrans : les pathologies à l'ère numérique*, L'échappée, 2019 ; Michel Desmurget, *La fabrique du crétin digital : les dangers des écrans pour nos enfants*, Seuil, 2019 ; Sabine Duflo, *Quand les écrans deviennent neurotoxiques : protégeons le cerveau de nos enfants*, Marabout, 2018.

classes) et le reste en *e-learning*. Comprendre, on mettra les élèves seuls devant leurs ordinateurs et des logiciels « intelligents ». Et on pourra couper dans ces effectifs si coûteux... pendant que les *Apple Distinguished Educators*, les *Microsoft Experts* et autres *Certified Google Educators* prendront les choses en main.

Où en sommes-nous ?

A son arrivée en 2017, le nouveau ministre de l'Éducation nationale, Jean-Michel Blanquer, a commencé par prendre une posture officielle plutôt distanciée avec le numérique. Le « grand plan numérique » a été discrètement remis : il s'agissait maintenant de bâtir « l'école de la confiance ». Le téléphone portable a même été interdit dans les écoles et les collèges pour la rentrée 2018, de quoi faire pousser quelques cris d'orfraie aux gardiens du temple numérique, certains moquant l'interdiction de ces « diaboliques équipements » et réclamant pas moins que... l'interdiction des stylos à l'école, sous prétexte qu'« [...] *en enseignant aux élèves à écrire avec un stylo plutôt qu'avec un logiciel de traitement de texte, nous ne leur enseignons pas les compétences dont ils auront besoin dans leur vie professionnelle et personnelle. Pour y remédier, les enseignants devraient encourager leurs élèves à écrire plus avec un logiciel de traitement de texte et moins avec un stylo* »¹⁵.

Mais pas d'inquiétude : si le téléphone portable est bien interdit dans la cour, en classe l'interdiction « *peut connaître des exceptions dans le cadre d'un usage pédagogique explicite et spécifique, encadré par les professeurs* »¹⁶. Ouf ! De fait, Jean-Michel Blanquer n'est pas technophobe, il est au contraire tout à fait fan des possibilités offertes par les « EdTechs » (*educational technologies*), les startups qui innovent au service de l'éducation – et notamment l'intelligence artificielle (IA) : possibilités pédagogiques, cela va sans dire, mais aussi possibilités de *business* pour un « écosystème » de jeunes pousses ou de grosses multinationales aux aguets de nouveaux marchés, voire possibilité de faire émerger, qui sait, en les soutenant par la juteuse commande publique, de futures *licornes* françaises, ambiance *start-up nation* au plus haut sommet de l'État oblige.

En juin 2018 a lieu l'inauguration du « Lab 110 bis », logé dans les bureaux du ministère, « espace dédié à l'innovation et favorisant l'intelligence collective ». Jean-Michel Blanquer y mentionne les axes désormais prioritaires du numérique à l'école : l'IA pour personnaliser les parcours, améliorer l'efficacité de l'enseignement, permettre au professeur de se concentrer sur l'essentiel, notamment par l'aide à l'évaluation ; le traitement massif des données pour permettre une meilleure orientation

15. Gilles Dowek, « Interdisons les stylos dès la rentrée prochaine », *Le Monde*, 05/11/2018.

16. <https://www.education.gouv.fr>.

des étudiants, proposer des formations plus pertinentes et améliorer l'employabilité ; les réseaux et plateformes pour réduire les inégalités spatiales ; et la robotique et les objets connectés pour permettre l'inclusion des plus fragiles.

Décryptons ce jargon, mille fois répété depuis des décennies : l'IA pour « personnaliser les parcours » et décharger le professeur du fardeau de l'évaluation consistera à coller l'élève devant son écran, et à mener des tests par QCM (questionnaire à choix multiple) vaguement améliorés sans doute, grâce à un peu de *machine learning* ; le traitement massif des données pour l'orientation, cela signifie que les parents et les futurs bacheliers n'ont pas fini de souffrir face à l'opacité du logiciel Parcoursup ; quant à la robotique pour les plus fragiles, mystère : peut-être s'agit-il d'exosquelettes pour certains enfants « plus fragiles » en cours de sport ?

Parfois, le jargon – et le ministre – se font plus lyriques. Ainsi à la journée de l'IA 2018 : « *Nous sommes en phase d'accomplissement, avec vingt-cinq siècles d'attente, de l'idéal socratique, celui de l'interactivité, celui d'une maïeutique réalisée réellement. Non pas parce que nous robotiserions l'éducation, non pas parce que nous la rendrions totalement technologique mais au contraire parce que nous réussirions une interaction grâce à un couple réussi entre l'homme et la machine. A mon sens, la question majeure de notre époque [...] c'est comment un monde de plus en plus technologique peut être un monde de plus en plus humain* »¹⁷.

Puis vint la crise sanitaire...

Avec la fermeture des écoles, la France effarée a d'abord découvert que l'instruction de nos enfants n'était pas leur seule fonction, mais que l'Éducation nationale remplissait aussi un rôle économique crucial, plus immédiat, celui d'*immense garderie*, pour les plus jeunes du moins, permettant aux parents de partir chaque jour au travail la conscience tranquille.

La grande accélération numérique, vécue dans tous les domaines, aurait pu être l'occasion de relancer le débat sur la place des outils digitaux à l'école ; d'autant plus que l'expérience de l'enseignement à distance, malgré l'effort à saluer de nombreux enseignants pour maintenir le lien avec leurs élèves malgré les circonstances, s'est révélé bien pitoyable. Las, le débat a surtout porté, dans les premiers temps, sur les inégalités d'équipement des élèves, tous n'ayant pas un ordinateur ou une tablette pour suivre les cours et rendre les devoirs à distance. Et si le contenu pédagogique des cours n'a souvent pas été à la hauteur, c'est évidemment, comme toujours, l'argument du manque de préparation, d'adaptation ou d'implication des enseignants qui a été brandi. Ainsi va la transformation numérique à marche forcée : si elle ne

17. <https://youtu.be/BKMwPrKYQV8>, à partir de la onzième minute ; relevé par Christophe Cailleaux, « La EdTech à l'assaut de l'éducation », dans Cédric Biagini, Christophe Cailleaux, François Jarrige, *op. cit.*, p.113.

tient pas ses promesses, c'est que les profs, toujours en retard d'un train, forcément, n'ont pas encore pris la pleine mesure de son potentiel pédagogique.

La crise a permis un « boom sans précédent du numérique éducatif »¹⁸, notamment dans l'enseignement à distance pour les entreprises. Pour ce qui est de l'éducation, c'est moins clair, car en France, « certains freins subsistent ». Pour Marie-Christine Levet, cofondatrice du fonds Educapital, start-uppeuse des EdTechs, s'il y a des progrès à faire, c'est dans le « *grand chantier [de] la digitalisation de notre éducation, de l'école primaire à l'enseignement supérieur. C'est là que nous avons le plus de retard* ».

Retard sur quoi, sur qui ? Peu importe ! C'est le principe même du *progrès* technologique : comme l'explique la reine rouge dans la suite des *Aventures d'Alice au pays des merveilles*, ne pas avancer (mais vers où ?), ce n'est pas faire du surplace, c'est reculer : « *Ici, vois-tu, on est obligé de courir tant qu'on peut pour rester au même endroit. Si on veut aller ailleurs, il faut courir au moins deux fois plus vite que ça. [...] Allons, allons, plus vite, plus vite !* »¹⁹.

A l'école du « monde d'après » ?

Le débat sur la numérisation généralisée de l'école n'a pas eu lieu. Il ne s'agit pas d'être technophobes, passéistes ou conservateurs, mais de regarder les faits le plus objectivement possible. D'ailleurs, pourquoi la charge de la preuve a-t-elle été inversée, pourquoi est-ce aux « détracteurs » du numérique de démontrer que celui-ci a plus d'impacts nuisibles que positifs ? Pourquoi acceptons-nous, collectivement, de prendre pour argent comptant les balivernes des vendeurs de matériel et de logiciel, ou leurs conseils et chercheurs inféodés, un petit cercle d'acteurs qui font carrière, dans le secteur public ou le secteur privé (voire les deux), sur les projets « innovants » autour du numérique ? Pourquoi faisons-nous *semblant de ne rien voir*, pourquoi refusons-nous de voir *ce qui crève les yeux* ?

La crise de l'école n'est pas née avec sa numérisation. Nous n'appelons aucunement à un retour à l'école d'antan, avec le tableau noir et la règle en bois, mais à faire preuve de « techno-discernement ». Quelles pourraient être concrètement les pistes de cette école du « monde d'après » ?

En premier lieu, il faut repenser la place des écrans dans les écoles ; il ne s'agit sans doute pas de les bannir entièrement (quelques usages ponctuels du numérique sont sans doute intéressants), mais tout est une question d'âge et de mesure. L'école pourrait être un « espace de désintoxication numérique »²⁰ pour nos enfants, jouer un

18. Ingrid Vergara, « Le boom sans précédent du numérique éducatif », *Le Figaro*, 29/03/2021.

19. Lewis Carroll, *De l'autre côté du miroir*, chapitre 2 [1871].

20. Jean-Pascal Gayant, « L'école doit être un espace de désintoxication numérique », *Le Monde*, 20/12/2020.

rôle d'apaisement, de protection, d'amortisseur de l'évolution de la société, au lieu de les livrer aux multinationales de l'informatique dès leur plus jeune âge. Plutôt que de courir après une éducation aux écrans qui devrait relever d'abord des parents, l'école, en s'appuyant sur ses enseignants, des livres papier et de l'innovation pédagogique véritable, pourrait offrir un temps de repos sanitaire et mental, au moins 30 heures par semaine à l'abri des ondes et de la « société du spectacle ». Il existe des expériences extrêmement intéressantes, comme les semaines sans écran, dont les effets bénéfiques et l'engouement suscité tant chez les élèves que dans leurs familles sont indéniables. Dans cette « école sans écran », le numérique aurait sa place, mais comme simple matière : on éduquerait *au* numérique et non *par* le numérique.

Il faut sortir du fantasme des *digital natives* : on ne naît pas digital, on le devient ! C'est l'entourage familial – et maintenant l'institution scolaire – qui offre aux enfants ces objets manufacturés et leur transmet son addiction. Nous rendons nos enfants physiquement dépendants du système numérique. Certes ils jonglent avec internet et les réseaux sociaux, mais ne savent pas comment sont fabriqués et fonctionnent les objets qui les entourent. On n'apprend plus de leçons par cœur, mais on accepte de sous-traiter sa connaissance et sa culture aux moteurs de recherche... On est loin de l'autonomie. Les promoteurs du numérique parlent de « faire tomber les murs de l'école », d'« habiter le monde ». Mais avant d'habiter le monde, il faut peut-être partir d'une base stable, commencer par comprendre son territoire. Les jeunes savent-ils où ils vivent, quelles plantes poussent dans les rues de leur ville, d'où vient leur eau potable, où partent leurs déchets ? L'école numérique, c'est un projet de déconnexion toujours plus grande de l'homme d'avec son milieu naturel.

En deuxième lieu, il faut reconcentrer l'école sur l'acquisition des fondamentaux (lire, écrire, compter) et de la culture générale, au lieu de s'acheminer vers une scolarité papillonnante – à l'image de notre société du « multitâches » et des cours de programmation introduits dès le primaire, pour apprendre le « langage informatique » avant même de maîtriser sa langue maternelle. Il existe une multitude d'innovations, d'idées non numériques, d'initiatives, d'expérimentations réussies, pour motiver les élèves, lutter contre les inégalités et la violence, vaincre le décrochage, retrouver même le goût de l'effort, au lieu de vouloir à tout prix « gamifier » l'enseignement.

Pourquoi, également, ne pas chercher un meilleur équilibre entre les matières académiques, techniques et créatives ? Le travail manuel a été dévalorisé, sa richesse intrinsèque, cognitive, sociale, psychologique a été dramatiquement occultée et dévalorisée depuis des décennies. La relation entre la main et le cerveau, fruit de millions d'années d'évolution, dans la construction de l'habileté mais aussi des savoirs, a été négligée. Renoncer à l'écriture, au profit du clavier ou de l'écran tactile – on n'en est pas encore tout à fait là, heureusement –, serait une énorme régression. Quant aux métiers de demain, quels seront-ils ? La seule manière de bien préparer les élèves est

de « bétonner » les fondamentaux et l'agilité intellectuelle, pas d'apprendre à coder dès l'école primaire...

Mais dans l'ambiance actuelle, il faudra du courage pour oser questionner la doxa techno-pédagogique, encore plus pour admettre que l'alternative non numérique est meilleure : « *Car contester la pensée digitale, c'est contester l'ordre social lui-même. [...] c'est quasiment faire la révolution. Et peut-être faut-il faire la révolution, c'est-à-dire avant tout se débarrasser des cuistres digitaux, des astrologues numériques et de tous les marchands d'algorithmes, qui savent tout, prévoient tout, règlent tout, pour que nous n'ayons qu'à accomplir, en ce bas monde, le seul acte héroïque d'un citoyen post-moderne : consommer* »²¹.

21. Jean-Pierre Despins et Marie-Claude Bartholy, *Arsenic et jeunes cervelles*, op. cit., p.207.



Mendeleïeva : un jeu de plateau pour mettre en valeur les femmes en sciences

Florence Sèdes ¹

1869 est l'année où le chimiste russe Dmitri Mendeleïev (1834-1907) a publié la classification périodique des éléments chimiques qu'il dispose selon leur masse atomique : le « tableau de Mendeleïev ² » est organisé selon un modèle qui permettait de prévoir les propriétés des éléments encore non découverts.

A l'occasion de ce 150^e anniversaire, des membres de l'association Femmes & Sciences ont conçu un jeu sur plateau mettant en valeur 125 femmes scientifiques qui ont travaillé ou travaillent au quotidien avec un des éléments du tableau. Ce jeu interactif permet, au travers de ces figures féminines, contemporaines ou historiques, de découvrir des disciplines, métiers ou sujets de recherche scientifique et des objets symbolisant l'élément, son contexte ou son utilisation.

Ces figures mises en lumière relèvent de sept domaines : biologie et santé, chimie, écologie et terre, énergie, mathématique et informatique, physique et astronomie, et sciences humaines et sociales (archéologie, anthropologie, philosophie...).

On trouve parmi elles des femmes aux carrières multiples, par exemple la chimiste Angela Merkel, la médaillée Fields Maryam Mirzakhani, mathématicienne, des « calculatrices de l'ombre » comme Katherine Johnson, des informaticiennes ayant marqué de manière décisive le domaine du *hard* ou du *soft* comme Alice Re-coque, à qui nous avons rendu hommage lors de sa récente disparition ³, ou Joëlle Coutaz.

1. Professeure d'informatique à l'université Toulouse 3 et chargée de mission Femmes et informatique à la SIF.

2. https://fr.wikipedia.org/wiki/Tableau_périodique_des_éléments.

3. <https://doi.org/10.48556/SIF.1024.17.119>.



Carte du jeu Mendeleïeva et boîte d'Alice Recoque correspondant à l'élément n°14 : silicium (Si). Les deux objets contenus dans la boîte représentent un microprocesseur et du sable. Ce dernier, particulièrement le quartz, contient un fort taux de silice sous forme de dioxyde de silicium (SiO_2) qui est l'ingrédient de base pour fabriquer des semi-conducteurs
(© Dominique Morello).

Au-delà des aspects scientifiques abordés, Mendeleïeva permet de parler des différents statuts et métiers de la recherche.

Ce jeu s'inscrit évidemment dans l'objectif de l'association Femmes & Sciences qui est de promouvoir les femmes en sciences et leur donner toute la place qu'elles méritent.

Leur invisibilité persistante minimise leur rôle dans l'avancée des connaissances et contribue à affaiblir l'attraction des jeunes filles pour les sciences.

Mendeleïeva constitue une ressource exceptionnelle de talents et d'expertises au féminin. Dans sa dernière version, ciblant plus spécifiquement les collégiennes et les collégiens, ce jeu met en lumière près de 140 femmes scientifiques de différents pays, contemporaines ou historiques, ordinaires ou extraordinaires, qui ont fait ou font leurs recherches sur un des éléments chimiques du tableau ou qui l'ont utilisé. Une version « adulte » est aussi disponible.

Il existe deux formes de jeu : le jeu *complet* constitue une sorte de cabinet de curiosités contenant 200 cartes à jouer présentant des femmes scientifiques, une paire de dés (à 10 faces), des objets se référant aux éléments chimiques (ampoule, tour Eiffel, pierre, circuit informatique...), le tout rassemblé dans des tiroirs en bois ou des boîtes en carton, numérotés de 1 à 99 (correspondant aux 99 premiers éléments chimiques du tableau).

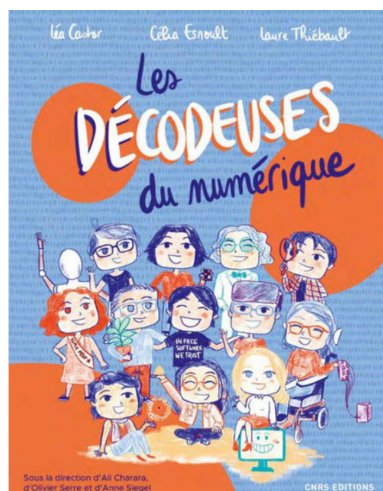
En l'absence du cabinet de curiosités, le jeu *simple* contient 200 cartes à jouer, deux dés à 10 faces et un tableau de Mendeleïev qui présente l'utilisation des éléments dans la vie courante. Il se joue en « jeu des 7 familles scientifiques ». Que ce soit dans la version complète ou dans la version simplifiée, le jeu consiste à lancer les dés et découvrir, par l'intermédiaire des cartes, les femmes scientifiques correspondant au nombre (à l'élément) obtenu, décrire leur métier et sujet de recherche, ainsi que les objets symbolisant l'élément ou son utilisation. Le jeu s'arrête lorsqu'une femme au moins a été trouvée et commentée dans chaque domaine. Le jeu peut se

jouer individuellement ou en équipe. Pour compléter le jeu, le public est invité à remplir un quizz qui permet d'aborder quelques aspects du tableau de Mendeleïev et de l'égalité femme/homme en matière d'éducation.



« Les décodeuses du numérique », une BD qui parle de, et fait parler de #femmesInformatique

Florence Sèdes¹



Douze portraits de chercheuses, enseignantes-chercheuses et ingénieures qui illustrent la diversité des métiers et des domaines de recherche en sciences du numérique : l'objectif de la bande-dessinée « Les décodeuses du numérique » est de contribuer à lutter contre les stéréotypes qui empêchent les femmes de s'engager dans cette voie.

Bien plus que le « code » ou l'intelligence artificielle, les sciences du numérique couvrent des champs aussi larges que les interfaces humain-machine, la robotique, le numérique frugal, l'analyse d'images ou de sons, la cybersécurité ou encore l'information quantique. Ces recherches résonnent étroitement avec les grands enjeux sociétaux de l'environnement, comme le changement climatique ou la transition énergétique, mais aussi la santé, la biologie, les villes du futur, ou la protection des données personnelles.

nement, comme le changement climatique ou la transition énergétique, mais aussi la santé, la biologie, les villes du futur, ou la protection des données personnelles.

1. Professeure d'informatique à l'université Toulouse 3 et chargée de mission Femmes et informatique à la SIF.

Au quotidien, les personnels de la recherche au CNRS et chez ses partenaires (organismes, universités et écoles d'ingénieurs) s'unissent pour faire avancer le front des sciences du numérique, mais les femmes sont encore minoritaires dans ce domaine. Le CNRS a fait de la politique parité-égalité l'une de ses grandes priorités, notamment dans les recrutements et l'évolution des carrières.

S'adressant à toutes et tous, jeunes, très jeunes ou moins jeunes, ludique, associant art et sciences, cet ouvrage veut inspirer les futures et futurs scientifiques et faire prendre conscience à toutes et tous des biais qui persistent.

Mettre en avant la diversité et la richesse des parcours, des motivations ou des hasards et rencontres qui mènent à une orientation, à une carrière, à un métier, dans ce domaine si riche et ouvert !

Les filles... OSEZ LE NUMERIQUE !

Publié par CNRS Editions², piloté par l'Institut des sciences de l'information et de leurs interactions du CNRS et illustré par Léa Castor³, l'ouvrage est disponible :

- en commande sur le site de CNRS Éditions ;
- en consultation libre⁴, avec des ressources supplémentaires, dans le cadre d'un partenariat avec différents acteurs de la médiation scientifique, la Société informatique de France, l'association Femmes et mathématiques, l'association Femmes et sciences et l'Onisep.

2. <https://ins2i.cnrs.fr/les-decodeuses-du-numerique>.

3. <https://www.instagram.com/leaa.caastor>.

4. <https://ins2i.cnrs.fr/les-decodeuses-du-numerique>.



La cordée de la réussite « décodeuses d'informatique »

Julien Narboux, Cristel Pelsser, Basile Sauvage et Gaelle Thomas¹

Le groupe de réflexion sur l'égalité homme-femme en informatique, à l'UFR de mathématique et informatique de l'université de Strasbourg, a fait le constat largement partagé que l'orientation dans les filières informatiques est très genrée, non seulement dans l'enseignement supérieur, mais aussi dès le lycée dans le choix de la spécialité « Numérique et sciences informatiques (NSI) ». Souhaitant favoriser un recrutement plus inclusif, nous nous sommes engagés dans le dispositif « les cordées de la réussite », au sein duquel nous avons organisé, dans les classes de collège et de lycée, des actions de médiation scientifique sur l'informatique et d'information sur l'orientation.

Les cordées de la réussite sont un dispositif national qui coordonne des actions pour aider les élèves dans l'élaboration de leur projet d'orientation. Ces actions visent en priorité les élèves des réseaux d'éducation prioritaire, des quartiers prioritaires de la politique de la ville ou de zones rurales. Ce dispositif — qui met en relation un établissement d'enseignement supérieur (dit « tête de cordée ») et des établissements d'enseignement secondaire — apporte à la fois un cadre institutionnel et des moyens de financement des différentes actions pour tous les participants.

Ainsi, depuis septembre 2020, l'UFR de mathématique et informatique est à la tête de la cordée intitulée « décodeuses d'informatique² », sur le thème « égalité

1. UFR de mathématique et informatique, université de Strasbourg.

2. Page web de la cordée : <https://mathinfo.unistra.fr/admission/lyceen/decodeuses-dinformatique>, wiki sur la médiation informatique à Strasbourg : <https://github.com/jnarboux/MediationInfoStrasbourg/wiki>.

des chances et égalité filles-garçons en informatique ». Cette cordée a pour objectif de clarifier, pour les élèves, ce qu'est la discipline informatique, et d'élargir les horizons concernant leur orientation future, notamment en luttant contre les stéréotypes de genre. Nous intervenons en groupe dans les classes en nous efforçant d'être au moins autant de femmes que d'hommes. Les ateliers sont animés en binômes constitués d'enseignants-chercheurs et d'enseignantes-chercheuses à l'université, de doctorants et de doctorantes, d'étudiants et d'étudiantes d'un niveau avancé en cursus informatique universitaire (L3 ou au-delà, spécialement formés pendant environ 12h).

Conformément aux règles des cordées et dans l'objectif de s'adresser aussi (et surtout) aux élèves qui ne sont pas spontanément intéressés par l'informatique, il n'y a pas de sélection parmi les élèves. Bien que quelques ateliers aient eu lieu à l'université, la règle générale est d'intervenir dans les établissements encordés, où les ateliers sont offerts pour tous les élèves d'un même niveau.

Nous animons des ateliers d'informatique débranchée, qui consistent à montrer des concepts d'informatique de manière ludique et sans utiliser d'ordinateur [JMV18]. L'avantage de cette approche est qu'elle permet d'aller loin dans les concepts tout en évitant les problèmes techniques. Matériellement, l'informatique débranchée s'avère bien adaptée au dispositif : il est facile d'organiser une intervention touchant plusieurs classes en parallèle sans contrainte de disponibilité des salles informatiques. De plus, le fait de manipuler physiquement des objets contribue à la motivation et l'implication des élèves — qui prennent plus de plaisir à participer aux activités — en insistant sur le fait que l'informatique n'est pas qu'une technique.

En classe de 4^e, nous proposons une animation type de deux heures par classe. Elle est constituée de plusieurs activités d'informatique débranchée, agrémentées de discussions pour travailler sur les représentations initiales et faire émerger les concepts informatiques sous-jacents (notamment les quatre concepts mis en exergue par Gilles Dowek : information, algorithme, langage, machine [Dow11]). Les questions de genre ne sont pas abordées explicitement, mais la présence d'animatrices informaticiennes aide à véhiculer une image mixte.

Certaines actions prévues en classe de seconde n'ont pas pu avoir lieu, mais, lors d'une prochaine édition, nous espérons pouvoir aborder plus directement les métiers de l'informatique, l'orientation, et les préjugés de genre.

Pour cette première année l'action a permis de toucher huit collègues totalisant environ 800 élèves.

Remerciements Cette action bénéficie indirectement du travail du groupe InforSansOrdi de la SIF et du groupe IREM Informatique de Strasbourg où nous puisons inspiration, suggestions et activités. Cette initiative est aussi inspirée de l'initiative L-codent L-crément de Philippe Marquet, Maude Pupin et Yann Secq.

Références

- [Dow11] Gilles Dowek. Les quatre concepts de l'informatique. In Georges-Louis Baron, Éric Bruillard, and Vassilis Komis., editors, *Sciences et technologies de l'information et de la communication en milieu éducatif : Analyse de pratiques et enjeux didactiques.*, pages 21–29, Patras, Greece, October 2011. Athènes : New Technologies Editions.
- [JMV18] Collectif Jean-Marc Vincent. *Tangente Éducation N°42-43 L'informatique débranchée.* Pole, 2018.



L'informatique de Claude Pair

L'apport de Claude Pair à la création de la science informatique (années 1963 – 1981)

Marion Créhange, Pierre Lescanne et Alain Quéré¹

Introduction

Sur une suggestion amicale de Claude Pair, nous avons accepté, 50 ans après, de présenter son activité scientifique en tant que l'un des fondateurs de l'informatique universitaire en France. Cet article couvre la période 1963–1981 : en effet après avoir été professeur en classe préparatoire, Claude rejoint l'enseignement supérieur en 1963. Dix-huit ans plus tard, en 1981, il s'oriente vers la haute administration en devenant directeur des lycées au ministère de l'Éducation nationale.

Avant d'aborder la rédaction, nous avons proposé à Claude l'organisation d'un colloque. Cette formule a permis de multiplier les témoignages et de mieux faire collectivement le bilan de ses recherches. Ce colloque a eu lieu au LORIA, à Nancy, le 14 juin 2019². Il a réuni 150 participants avec 17 intervenants dont Antoine Petit, PDG du CNRS. Près de deux ans plus tard, nous avons assemblé nos notes, puisé dans les documents à notre disposition et activé notre mémoire.

1. marion.crehange@sfr.fr, pierre.lescanne@ens-lyon.fr, apmf.quere@laposte.net. Article également disponible sur HAL, <https://hal.archives-ouvertes.fr/hal-03193950v2>.

2. <http://claudelpair.fr/>.

Précisons d'entrée que le lecteur ne trouvera pas ici des démonstrations rigoureuses ni des formalisations complètes. Il lui faudra pour cela consulter les nombreuses publications de Claude ou les 48 thèses qu'il a dirigées³.

Nous avons choisi d'ajouter à des propos scientifiques des parties plus anecdotiques signalées en italique. Nous espérons ainsi aérer la lecture et évoquer l'ambiance du travail d'équipe que Claude a instauré, équipe qui au départ se comptait sur les doigts d'une main. Équipe interuniversitaire qui, en grandissant, sera associée au CNRS en 1973 grâce à Claude, puis constituera le Centre de recherche en informatique de Nancy (CRIN) en 1975. La croissance s'est poursuivie : 70 personnes en 1977, une centaine en 1984 avec la venue de l'INRIA⁴.

Il faut se souvenir qu'au début des années 60, la science informatique est dans la petite enfance. La plupart des domaines que nous connaissons aujourd'hui sont alors à créer, le plus souvent dans l'indifférence — ou même contre l'avis — des mathématiciens qui ne voient alors aucune raison de s'intéresser aux ordinateurs. Il faut aussi rappeler que les moyens de communication actuels n'existent pas. Il n'y a pas encore de terminologie propre à la recherche informatique⁵ et beaucoup de chercheurs français publient dans leur langue maternelle, ce qui contribue à les isoler dans le monde. C'est dans ce cadre que Claude Pair imagine des concepts qui ont été redécouverts par la suite. Comme nous le montrerons, Claude est un bel exemple d'application de la loi de Stigler⁶ qui affirme qu'une découverte scientifique ne porte jamais le nom de son premier auteur. Nous souhaitons donc que le lecteur retienne de ce texte le caractère extrêmement novateur de ses recherches. Le recul du temps permet en effet d'affirmer qu'il a été un authentique visionnaire.

Les débuts : la compilation

Vers l'informatique

Comme sa formation initiale l'y conduit⁷, Claude devient professeur de « taupe » à Metz avant son service militaire qui lui offre l'opportunité de faire ses premiers pas en « calcul automatique » au CEA en 1958-59. Il y programme en langage machine

3. Pour être précis, 47 thèses sur la période considérée ici puis, quelques années plus tard, celle de Radhia COUSOT (1985).

4. En 1997, l'ensemble est devenu le LORIA, unité mixte associant CNRS, INRIA et les universités. En 2017, il comptait 300 permanents de 48 nationalités (rapport HCERES). Sur l'histoire de l'informatique à Nancy, voir aussi [9, 25].

5. Un seul exemple : pour traduire l'inélegant *garbage collector*, Claude a introduit le terme « ramasse miettes », voir Wikipedia.

6. Il se dit que cette loi s'applique aussi à son auteur, voir « loi de Stigler » sur Wikipédia.

7. Né à Blâmont (Meurthe-et-Moselle) en 1934, Claude devient un élève brillant au collège à Lunéville. Il est admis en 1951 au lycée Louis le Grand à Paris, puis en 1953 à l'École normale supérieure. Il est agrégé en 1956, et nommé professeur de mathématiques spéciales à Metz (1956-1957).

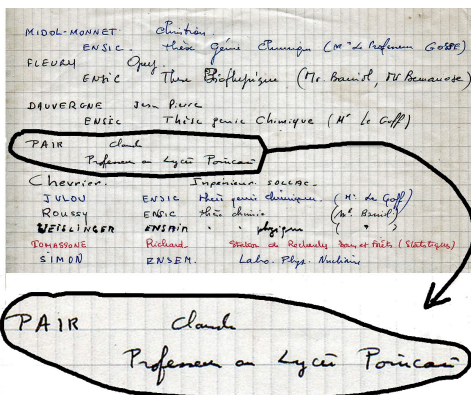


FIGURE 1. Émargement au cours du soir d'informatique en 1963.

sur Bull Gamma « à extension tambour », concurrent de l'IBM 650. Il est ensuite nommé à Nancy, au lycée Henri Poincaré.

À Nancy, l'informatique universitaire a été introduite par Jean Legras⁸ dès 1957 avec une machine à programme câblé, l'IBM 604, puis un ordinateur⁹ IBM 650 dès 1958¹⁰. En 1962, Claude prend contact avec Jean Legras qui lui donne accès au « Centre de calcul » qu'il a créé, et l'invite à suivre à la rentrée le cours de programmation sur 650 destiné à des adultes (cf. figure 1). Marion Créhange, chargée de la seconde partie, ne se doutait pas que cet élève, qui lui paraissait doué, allait devenir le père de la recherche en informatique à Nancy et serait son patron de thèse d'état!

Claude prend vite conscience que, derrière des machines différentes, se cache une réalité commune qu'il préfère explorer, plutôt que de s'orienter vers l'analyse numérique, spécialité de Jean Legras. Au Centre de calcul, il consulte la collection des Communications de l'ACM et d'autres revues. C'est ainsi qu'il découvre un nouveau langage de programmation, Algol [3], défini par un groupe international indépendant de tout constructeur, donc attirant pour des universitaires, et beaucoup plus

8. http://www.professeurs-medecine-nancy.fr/livre_jean_legras.htm.

9. Le mot « ordinateur » a été inventé en 1955 par le philologue Jacques Perret, en réponse à une demande d'IBM, qui voulait un nom plus parlant que « calculateur » pour son IBM 650.

10. L'Est Républicain du 6 novembre 1958 annonce la création d'un institut de calcul automatique dont la direction sera confiée à Jean Legras : « Doté d'un matériel ultra moderne, cet institut est destiné à être commun à toutes les facultés de la région de l'Est; les cours seront notamment suivis par les élèves de 3ème cycle de mathématiques appliquées. On pense qu'il pourrait ouvrir ses portes dans un délai de deux mois : au 1er janvier très probablement ».

novateur que le Fortran d'IBM : la récursivité de la définition des programmes et des fonctions fait rêver. Un programme n'est plus une suite d'instructions apparaissant au même niveau, mais commence à ressembler à un texte en langue naturelle ; sa syntaxe est d'ailleurs décrite par des règles proches des grammaires formelles dites *context-free*¹¹, introduites quelques années plus tôt par le linguiste Noam Chomsky. Compiler ce langage demeure un problème sérieux. Claude, devenu pour un an attaché de recherche au CNRS, constitue une équipe dans ce but et, dès 1963, il signe une première publication aux Comptes-rendus de l'Académie des sciences [P1], puis, en 1964, il publie l'article intitulé « Arbres, piles et compilation » dans la revue française de traitement de l'information [P2].

Cours de DEA, création de l'équipe

En 1963, Jean Legras assiste, à Grenoble, à un séminaire dont l'objet est la présentation de la version française de la définition d'Algol 60. Claude se souvient de son retour enthousiaste : « un langage avec des SI et des DO! ». Jacques André¹²[T3] raconte : « toute affaire cessante, ils ont organisé dans le cadre du DEA un cours sur Algol 60 : ça a été pour moi une révélation car c'était quand même plus passionnant que le PASO de la 650! Quand Claude a parlé au début de l'année scolaire suivante de prendre des étudiants en 3^e cycle pour faire un compilateur d'Algol 60, j'ai sauté sur l'occasion ».

À cette époque, Claude met en place des réunions régulières, premier « séminaire informatique nancéien », associant théorie et applications. Il signe plusieurs publications internes au Centre de calcul.

Jacques André raconte : « Claude Pair nous donnait des conférences sur les langages et nous devions chacun rédiger un chapitre. Je me souviens qu'il m'avait rendu ma prose couverte de rouge. Voyant cela, je m'étais dit que ce n'était pas la peine que je continue... Mais en réalité il était content de ma rédaction et visait seulement à en améliorer le fond ».

Compilateur Algol 60 (1963-1965) et thèses à Nancy

Puisqu'il n'existe pas encore de cursus informatique, la plupart des étudiants qui rejoignent cette équipe jusqu'en 1968 viennent des « mathématiques pures ». C'est le cas de Michel Cusey [T1], premier thésard de Claude et pilote de la petite équipe.

Jacques André se souvient : « On a commencé le compilateur sur IBM 650, j'y ai travaillé la lecture des nombres, et assez vite, bonne nouvelle, on allait avoir accès à un 1620 chez IBM (cf. figure 2). Mais cette machine était à Metz et n'était disponible qu'après la fermeture des bureaux. Tous les quatre¹³, on

11. Que Claude, adepte de la francophonie, rebaptisera « langages à contextes libres » dans son cours.

12. <http://jacques-andre.fr/>.

13. Jacques André, Michel Cusey, Alain Floc'h, Jean-Marie Laporte.



FIGURE 2. IBM 1620, source Wikipédia.

y allait avec la voiture de Michel, départ vers 17 heures et retour... quand nos essais étaient finis, le plus souvent à deux heures du matin. La bécane était en vitrine et on avait l'impression d'être des ours en cage quand les passants nous regardaient avec curiosité. On « sandwichait » autour de la machine jusqu'au jour où le directeur du centre IBM nous a convoqués pour nous signifier assez rudement que les bières sur le pupitre, ce n'était pas vraiment dans l'esprit de l'entreprise. Du coup, on est allé souvent prendre une choucroute à la fin du travail à la brasserie de la gare de Metz, et je n'en ai jamais mangé d'aussi bonnes depuis... ».

Avant d'envisager la programmation du compilateur, Michel Cusey travaille sur son ossature générale : l'analyse syntaxique, les déclarations de variables, la gestion des piles et de l'ensemble de la mémoire, la construction du programme objet. Il le fait en respectant scrupuleusement les idées de Claude qui souhaitait vérifier sur un cas concret la pertinence de son approche mathématique. Claude est ravi que sa théorie « marche » et il lui en est reconnaissant. Jacques André est en charge des entrées-sorties et des procédures de service. La définition du langage Algol 60 ne définissant rien sur les entrées-sorties, il adopte les propositions de Donald Knuth [21]. Jean-Marie Laporte s'occupe des tableaux. Plus tard, Marion Créhange traite des procédures, en particulier de la récursivité, avec la collaboration d'Alain Floc'h qui sera ensuite chargé de la mise au point finale du compilateur.

Citons Claude¹⁴ : « En 1965, le compilateur est presque achevé. Le groupe européen des utilisateurs de 1620 me propose de le présenter à sa réunion de Mannheim [P5]. À partir de là, l'université de Saint-Andrews (Écosse) me demande de le lui envoyer sur cartes perforées, l'outil de l'époque pour entrer des informations dans un ordinateur. Mais, patatras, pour remplacer le 650 vieillissant, Legras choisit, dans le cadre du Plan calcul naissant, plutôt que le

14. A tout CRIN : de la naissance à la maturité (1963-1976), Archives Poincaré, Nancy, janvier 2019.

1620, une machine française CAE 510 qui possède déjà un compilateur Algol. Notre motivation disparaît et IBM arrête son prêt, avant que le compilateur soit réellement utilisable par d'autres que nous ».

Dans les mois qui suivent, l'équipe commence à se disperser. Mais l'expérience ne sera pas perdue. Des notions ont été dégagées, comme celles d'analyse syntaxique et de pile, des étudiants ont été formés, en particulier à la théorie des langages, et cinq thèses de troisième cycle ont été soutenues, dont trois à propos du compilateur [T1, T2, T3, T4, T5].

Fin 1965, Claude soutient sa thèse d'État, « Étude de la notion de pile, application à l'analyse syntaxique ». Elle récapitule les méthodes d'analyse syntaxique des langages de Chomsky, mettant en évidence trois familles d'algorithmes de construction d'arbres syntaxiques, une ascendante et deux descendantes « en profondeur d'abord » et « en largeur d'abord », dira-t-on plus tard. D'un style formel, rangée dans la spécialité « mathématiques », elle suscitera quelque incompréhension des mathématiciens « purs ». Par la suite, Claude pensera d'ailleurs que sa présentation était trop mathématique : « *Il faudra un peu de temps pour inventer un style d'écriture rigoureux mais moins uniquement mathématique, adapté à cette nouvelle science et tenant compte de ses composantes fondamentales : algorithmes [P66], types de données [P24] (comme les piles ou les arbres syntaxiques d'une grammaire de Chomsky), langages de programmation* ». Quoi qu'il en soit, il s'agit de la seconde thèse d'État en informatique, après celle que Jean-Claude Boussard avait soutenue l'année précédente à Grenoble, décrivant un compilateur Algol qu'il avait écrit pour un IBM 704 que son université avait la chance de posséder. Cette thèse, elle, était qualifiée « de science appliquée » car l'informatique n'était pas encore reconnue comme une discipline à part entière à l'époque dans les universités françaises.

Le chercheur et sa prescience

L'analyse syntaxique de précedence

L'analyse syntaxique est la première étape de la compilation : elle consiste à calculer, d'après la chaîne de caractères du programme à traduire, sa structure grammaticale sous la forme d'un arbre. Cette opération doit être faite de façon déterministe si on veut éviter tout retour en arrière. Claude montre que c'est possible pour le langage Algol, en mettant au point un algorithme (cf. figure 3) qui le permet, fondé sur une analyse ascendante, qui sera celui du compilateur nancéien et qu'il décrit dans un article publié en 1964.

Cet algorithme, redécouvert en 1966 par Niklaus Wirth et Helmuth Weber [36], est nommé méthode de « précedence » (ordre de priorité sur les caractères au cours de l'analyse). Dans leur livre sur l'analyse syntaxique [1], Aho et Ullman parlent du concept de précedence de Wirth-Weber (p.404), mais,

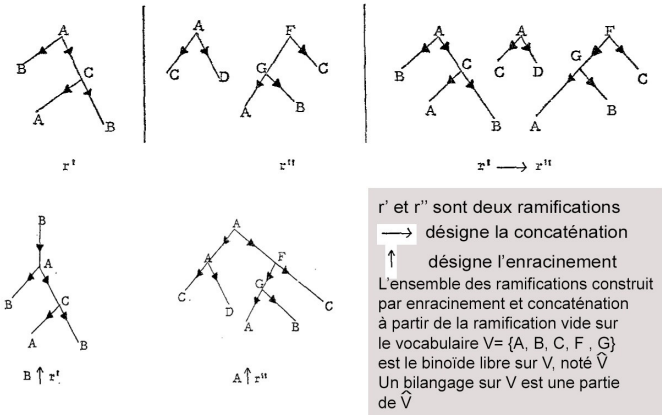


FIGURE 4. Ramifications et bilangages (*Information and Control* [P11]).

Langages et bilangages

Claude est donc féru d’algèbre, et c’est pourquoi la théorie des langages le comble. Un langage est un ensemble de phrases, éléments unidimensionnels ; mais l’analyse syntaxique construit des arbres, à deux dimensions, et Claude nomme « bilangage » un ensemble d’arbres (cf. figure 4). De même qu’un langage a une structure algébrique de monoïde (une seule opération), les bilangages conduisent à deux opérations, pour engendrer les arbres, et à la structure de « binoïde ».

L’histoire commence en mai 1968, par la publication de l’article de Claude Pair et Alain Quéré [T8] « Définition et Étude des Bilangages Réguliers » dans la revue Information and Control [P11]. Le mot « régulier », utilisé pour les langages reconnus par un automate fini, est étendu ici aux bilangages : l’ensemble des arbres syntaxiques d’un langage context-free est un bilangage régulier.

Après cette publication, Claude reçoit de nombreuses demandes de tirés à part. L’une vient de Rózsa Péter une chercheuse hongroise, pionnière de la calculabilité et qui a écrit le tout premier livre sur les fonctions récursives [29] et publié sur elles des contributions remarquables.

En particulier, en utilisant la récurrence structurelle, Rózsa Péter a eu l’idée d’étendre la récursivité qui, à l’origine, se limitait aux entiers naturels, à ce que nous appellerions aujourd’hui des structures de données bien fondées. Elle les nomme *angeordneten freien holomorphen Mengen* (ensembles libres holomorphes structurés).

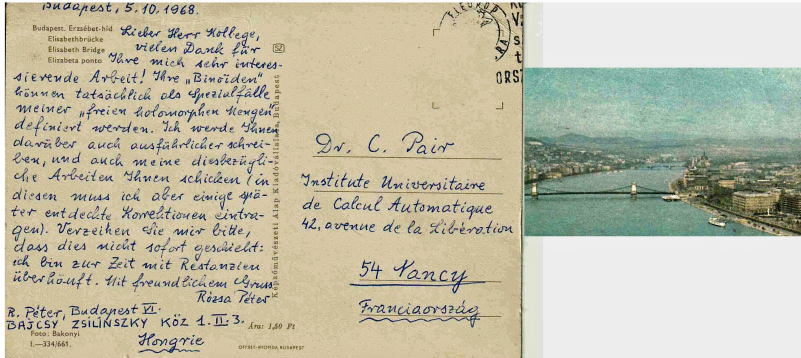


FIGURE 5. Rózsa Péter accuse réception du tiré à part et annonce sa réponse sur le fond.

On peut comprendre son excitation quand elle a vu qu'à partir de besoins de formalisation en informatique, Claude proposait une telle structure pour les arbres syntaxiques. Claude envoie donc aussitôt l'article qu'elle demande et le remerciement arrive par une jolie carte postale du Pont Elisabeth à Budapest (cf. figure 5).

Elle se plonge donc avec passion dans les écrits de Claude, y passant des nuits, lisant et réfléchissant jusqu'à 4–5 heures du matin, comme elle l'écrit avec enthousiasme et fraîcheur, dans sa lettre du 28 octobre 1968. On a l'impression qu'à travers le rideau de fer qui divise alors l'Europe, elle parle à un ami qu'elle connaît depuis toujours, bien qu'elle ne l'ait jamais rencontré. Elle annonce dans cette lettre l'article qu'elle va publier dans Acta Mathematica dont elle envoie, à son tour, un tiré à part à Claude, article intitulé : « Die Pairschen freien Binoïden als Spezialfälle der angeordneten freien holomorphen Mengen ». On notera la création, en allemand, du nouvel adjectif « Pairschen » !

En réponse, Claude et son élève Alain Quéré publient dans la même revue, à l'invitation de leur collègue hongroise, un complément intitulé « Sur les fonctions récursives primitives de ramifications » qui montre que l'égalité est primitive récursive dans le binoïde libre qu'on peut comparer aux ensembles holomorphes libres de Rózsa Péter.

Dans la ligne de recherche ainsi créée, en 1971, Pierre Lescanne [T16] a emprunté à la théorie des catégories ses concepts pour généraliser les langages réguliers, reconnaissables par automates et solutions d'équations spécifiques. Il montre ainsi que, même généralisées, ces trois familles de langages coïncident.

En 1974, Pierre Marchand étudie les « bigrammaires » et les systèmes transformationnels [T26]. Plus tard, en utilisant des algèbres libres (objets mathématiques à plusieurs opérations, dont les chaînes de caractères ne sont qu'un cas très particulier), il généralise, lui aussi, les notions de langages reconnaissables et de langages algébriques, mais dans un cadre plus général où des relations existent entre les opérateurs. Il montre la construction algorithmique des algèbres minimales de reconnaissance pour les parties reconnaissables, et, par ailleurs il explique, par l'adjonction d'une loi à une algèbre, la hiérarchie de Chomsky, qui devient alors infinie [T45].

L'équivalence des types infinis

Le groupe Algol francophone, que les collègues grenoblois de Claude lui ont proposé d'animer, se fixe pour objectif de faire connaître le nouveau langage Algol 68, notamment à travers plusieurs publications comme la traduction du rapport de définition et un copieux manuel d'utilisation. Algol 68 fait avancer la science informatique : le principal progrès est que les types de données n'y sont plus prédéfinis comme en Algol 60 (nombres entiers, réels, valeurs booléennes, tableaux), mais peuvent être introduits en fonction des besoins de chaque programme. Il permet donc de définir des types de données, et de le faire éventuellement de manière récursive.

La question se pose alors de savoir si deux définitions distinctes sont équivalentes, c'est-à-dire conduisent au même type. En 1970, Claude publie un algorithme de décision dans la revue Algol Bulletin, alors destinée aux spécialistes d'Algol 68, sous le titre un peu imprécis *Concerning the Syntax of Algol 68* [P19].

Cette décidabilité est redécouverte en 1993 dans un cadre plus général par Roberto Amadio et Luca Cardelli [2]. La démonstration repose sur un concept mis en évidence autour de 1980 : la coinduction. Cardelli [2] a indiqué que, cherchant une antériorité, il avait pensé à Simula, mais n'avait pas regardé du côté d'Algol 68. L'aurait-il fait, qu'il n'aurait sans doute pas trouvé l'article de Claude, bien caché dans la littérature scientifique!

Aujourd'hui, nous dirions que la démonstration de la décidabilité d'un problème se fait soit par réduction à un autre problème décidable, soit par la description d'un algorithme de décision, et c'est ce que propose Claude dans la section 4 de l'article cité [P19]. Il propose une méthode de raisonnement sur des objets infinis qui, comme plus tard la coinduction, fait appel à la résolution d'une équation à point fixe¹⁵. Une équation à point fixe peut avoir plusieurs solutions. Alors que l'induction s'appuie sur un raisonnement fondé sur le plus petit point fixe, la coinduction s'appuie sur le plus grand point fixe. Claude ne peut pas connaître ces différences, qui ne seront mises en évidence qu'une décennie plus tard, mais il doit en avoir l'intuition, car

15. Une équation à point fixe est une équation de la forme $X = F(X)$. Si l'ensemble dans lequel on cherche les solutions est ordonné, par exemple par inclusion d'ensembles, il y a une plus grande et une plus petite solution (on dit aussi un plus grand et un plus petit point fixe).

sa démonstration fonctionne en deux temps. Premier temps, il démontre l'unicité du point fixe, le plus petit point fixe est aussi le plus grand. Deuxième temps, il démontre que sur cet unique point fixe, on peut mettre en œuvre un processus de décision.

Et Benjamin Pierce peut donc écrire dans son livre de référence [31] qu'il s'agit bien de la première démonstration de la décidabilité de l'égalité de deux types (équi)-récurifs alors qu'il lui faut 648 pages pour présenter aux étudiants en master la notion que Claude avait introduite, à savoir les types¹⁶ dans les langages de programmation.

Les structures de données

En juillet 1971 à Alès, lors de la première école d'été de l'AFCEC¹⁷, Claude donne l'un des trois cours, « les structures de données et leur représentation en mémoire ». Ce cours, destiné à la programmation, sera revu par Marie-Claude Gaudel et publié par l'IRIA en 1977 [P24].

Une donnée est un objet informatique (liste, pile, arbre, table...) susceptible d'être transformé par des opérations. Une structure de données [P33] indique ces opérations et leurs propriétés, de la même façon qu'en mathématiques une structure algébrique (groupe, anneau, corps...) comprend une ou plusieurs opérations et leurs propriétés sous la forme d'axiomes.

En 1972, Claude obtient du CNRS un contrat d'Action thématique programmée (ATP) sous le titre « Informatique théorique, programmes et données ». Il le confie à Jean-Luc Rémy qui précise en particulier que les axiomes sur les opérations seront des égalités (axiomes équationnels). Le contrat dure deux ans et fait l'objet en 1974 d'un rapport de Claude au nom de l'équipe de recherche associée 364 au CNRS et de l'université de Nancy II.

Dans les années qui suivent, Claude commence à s'effacer devant ses chercheurs qui prennent leur envol, échangent entre eux et publient ensemble. Jean-Pierre Finance et Jean-Luc Rémy font une communication à l'école d'été de Grenade (1973, [13]), à la suite de quoi Jean-Pierre Finance reprend avec Claude la rédaction d'un document de travail qui donnera lieu à un article dans une des premières conférences IFIP [P30]. Deux thèses en sont issues, soutenues toutes les deux le 25 juin 1974 : celle de Jean-Luc Rémy [T24] étudie les aspects les plus formels, par une utilisation de la logique mathématique ; la thèse de Jean-Pierre Finance [T25] utilise les structures de données pour décrire la sémantique d'un langage de programmation. Plus tard, celle de Marie-Claude Gaudel [T43] emploiera la sémantique pour engendrer des compilateurs et prouver leur correction. En 1979, dans sa thèse [T39], Pierre Lescanne revisite les structures d'information dans une approche d'algèbres, préfigurant les contributions de Yuri Gurevich et de ses collaborateurs.

16. Qu'il appelle « mode » à la mode de l'époque !

17. La société savante d'informatique à l'époque.

Indépendamment de l'équipe nancéienne, des développements très similaires ont émergé à partir de 1993, notamment grâce à Yuri Gurevich, sous le nom d'Evolving Algebras [18], puis de Sequential abstract-state machines, nom plus attractif [19], quand l'industriel Microsoft s'y est intéressé.

Ces recherches sur les données ont stimulé une activité de nature didactique sur la théorie des programmes et l'informatique théorique en général, qui sera le support d'un cours à l'école d'été d'informatique de Tarbes en 1974 puis conduira à l'écriture d'un livre publié en 1978 chez Dunod [26].

Entre eux, les sept auteurs Jean-Pierre Finance, Monique Grandbastien, Pierre Lescanne, Pierre Marchand, Roger Mohr, Alain Quéré et Jean-Luc Rémy, s'appelaient, dans un premier temps, Blanche Neige, nom inapproprié, puisqu'ils n'étaient que sept nains, sous le charme d'une princesse. Le nom de Livercy (pour Liverdun et Commercy, deux villes lorraines où eurent lieu les séances d'édition) fut finalement retenu. Ce livre, l'un des premiers à traiter d'informatique théorique, toutes langues confondues, fut utilisé, par exemple, par Jean-Louis Lassez comme base de ses cours à l'Université de Melbourne en Australie ou par Olivier Danvy, professeur à l'Université d'Aarhus au Danemark, pour s'initier à la théorie des continuations.

Autres approches et nouvelles pousses

Claude aime aussi étendre son domaine de compétence et il ne se limite pas à un seul axe en matière de recherche. Nous évoquons ici la diversité de ses intérêts. Cependant, nous sommes loin d'un inventaire à la Prévert, car, chez Claude, ses intérêts sont tous ancrés sur une exigence de rigueur scientifique. Parcourons brièvement ces autres approches qui sont devenues aussi de nouvelles directions de recherche au CRIN.

En informatique, dans la lignée des premières recherches, au début de la décennie 1970, les compilateurs de compilateurs se multiplient, le plus connu étant le couple *lex* et *yacc*. Le nom de *yacc*, « encore un compilateur de compilateur » montre combien le sujet est chaud à cette époque. Sa version finale est publiée en 1975 [20]. À Nancy, Claude aiguille Françoise Bellegarde [T17] et Jean Maroldt [T18] sur ce sujet, ils soutiennent une thèse en 1972. Notez l'antériorité du *yacc* nancéien, utilisé aussi comme outil d'enseignement. Il s'appellera Fabrication automatique de compilateurs efficaces (Face) et mis à profit pour la mise au point de compilateurs (Anne-Marie Rasser [T30]).

Dès les débuts, Claude avait conscience que l'informatique, science transversale, pouvait servir en dehors de son champ propre et donner lieu à des recherches mixtes, au-delà de simples utilisations. La liste des thèses dirigées par Claude et

explorant des nouvelles branches donne un aperçu de cet esprit d'ouverture. Indiquons quelques exemples qui montrent l'étendue des domaines abordés : linguistique [T9, T21, T22], médecine [T11, T19], enseignement [T42], représentation des connaissances [T46].

Le plus souvent ce sont les thésards eux-mêmes qui ont choisi des orientations correspondant à leur compétence. Ils auraient pu être découragés par un patron trop directif. Au contraire, Claude les a encouragés dans le sens de leurs goûts. Donnons quelques exemples.

Jean-Claude Derniame s'est intéressé à la prise en compte dans un même environnement des différentes étapes de construction du logiciel : spécification, analyse, programmation, exécution, maintenance. En référence au dieu hindou auquel la tradition attribue cinq bras, il nomme CIVA son projet qui traite de définition modulaire et de classes d'objets, préfigurant les langages à objets si populaires aujourd'hui [T23].

Pour Marion Créhange, c'est un stage au laboratoire d'Analyse documentaire pour l'archéologie (CNRS, Jean-Claude Gardin à Marseille) qui la détermine à s'orienter vers les questions de recherche d'informations. Ensuite, elle collabore avec la médiéviste Lucie Fossier qui l'avait sollicitée pour traiter des documents diplomatiques du Moyen Âge au Centre de recherche et applications linguistiques (CRAL) [15]; ce qui lui permet d'acquérir une expérience en recherche documentaire. Plus tard, ce sera la conception de PIVOINES, langage d'interrogation de bases de données, très « déclaratif » et indépendant des choix de représentation [T29] dans la ligne des études théoriques sur les structures de données. Voilà un exemple représentatif des allers-retours entre théorie et application qui se fécondent l'une l'autre, une démarche qui intéressait particulièrement Claude¹⁸.

Autre exemple : Maryse Quéré [T42] avait débuté sa recherche en mathématiques pures, puis s'était consacrée à la formation d'adultes au CUCES¹⁹. Elle avait ensuite participé à celle de professeurs de lycées à l'informatique, puis créé le Centre lorrain d'enseignement par ordinateur (CLEO), d'où son attrait pour l'usage innovant de l'informatique dans l'enseignement. Pour Roger Mohr²⁰, la proposition de Claude d'imaginer les figures comme des objets à deux dimensions engendrés par une grammaire [T20] a été un premier départ vers la science des images, à laquelle il s'est ensuite consacré en

18. Marion écrit « j'ai pu apprécier la qualité extrême de Claude comme directeur de thèse, ferme et incitatif, mais gentil et promouvant, préoccupé par la réussite de ses élèves. »

19. Autre institution nancéienne, le Centre universitaire de coopération économique et sociale et l'un des tous premiers organismes de formation des adultes, créé par le recteur Capelle et André Grandpierre de la société Pont-à-Mousson, puis dirigé par Bertrand Schwartz.

20. Cf. *In memoriam Roger Mohr*, revue 1024 n° 11, septembre 2017.

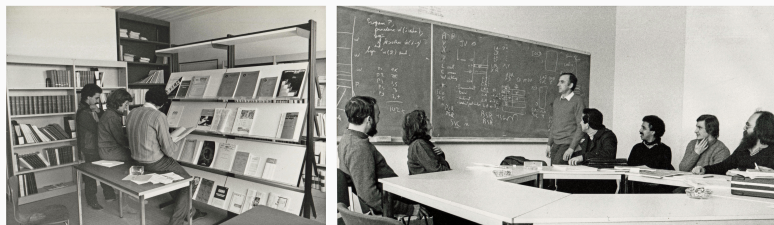


FIGURE 6. Bibliothèque et salle de réunion au CRIN en 1982.

créant au CRIN, avec Jean-Paul Haton, l'équipe Reconnaissance des formes et intelligence artificielle (RFIA).

Recherche et enseignement

Didactique de la programmation : la méthode déductive

Pour Claude, l'informatique naissante n'est pas seulement un objet de recherche, c'est aussi une discipline à enseigner. Dans les années 60, on sait laborieusement expliquer un programme, mais la démarche qui consiste à trouver un algorithme pour résoudre un problème reste difficile, voir impossible, à expliciter. Le plus souvent l'enseignant, tel un magicien, dessine un organigramme qu'il transpose ensuite dans un langage de programmation. Si la seconde partie du travail est facile, l'invention de l'algorithme est certes immédiatement saisie par les étudiants les plus brillants, mais reste incompréhensible pour la plupart. La conséquence de cette situation est l'erreur de programmation, le plus souvent découverte lors d'une exécution sur machine et rectifiée par des essais successifs sans s'appuyer sur des preuves sérieuses. Depuis 1968, la programmation structurée [12], [10] a mis en évidence une construction raisonnée des algorithmes et il existe quelques méthodes, le plus souvent sous forme de recettes standard pour résoudre certaines familles de problèmes (Méthode Warnier [34]).

D. E. Knuth propose « la programmation littéraire » [22] qu'on peut rapprocher de la méthode que Claude introduit à partir de 1973. Cette méthode sépare par étapes les activités de définition du problème, d'écriture de l'algorithme et enfin du programme. L'idée de départ est aussi simple que nouvelle : résoudre un problème, c'est partir du but à atteindre, « le résultat », et remonter le chemin jusqu'aux entités connues, appelées « données ». Autrement dit, construire un programme, c'est raisonner à partir de la définition du résultat pour revenir aux données en utilisant à chaque étape une définition puisée dans une liste prédéfinie de primitives usuelles de l'algorithmique : formules, conditionnelles, itérations — considérées ici comme

la définition de suites —, fonctions, dont la définition apparaît comme un nouveau problème à résoudre. Chaque étape de la construction est une déduction, d'où le nom de « méthode déductive ».

Pour illustrer la démarche contentons-nous d'un exemple simplissime, à savoir la recherche du maximum de 3 nombres. Commençons donc par écrire la caractérisation informelle du résultat :

$$M : \text{est le maximum de } a, b, c$$

Il s'agit alors de choisir, parmi les définitions algorithmiques référencées dans la méthode, l'une d'entre elles qui nous fera progresser vers la solution en introduisant au besoin de nouvelles entités. Ce sera, ici, une définition conditionnelle :

$$M = \text{ si } a > mbc \text{ alors } a \text{ sinon } mbc$$

avec introduction de deux intermédiaires à définir :

$$a = \text{ donnée,}$$

$$mbc = \text{ maximum de } b \text{ et } c$$

et l'introduction de deux intermédiaires à définir, b et c qui sont des données. Une nouvelle conditionnelle porte sur des données qui sont connues :

$$mbc = \text{ si } b > c \text{ alors } b \text{ sinon } c$$

Ce qui résout le problème. La méthode organise ce travail dans une table à deux colonnes, les définitions informelles et définitions formelles. Ici :

Définitions informelles	Définitions formelles (algorithmiques)
M : Maximum de a, b, c	$M = \text{ si } a > mbc \text{ alors } a \text{ sinon } mbc$
mbc : maximum de b et c	$mbc = \text{ si } b > c \text{ alors } b \text{ sinon } c$
a, b, c : données	$a, b, c = \text{ lire}$

Pour faire de ces définitions un algorithme séquentiel, il reste à les organiser, c'est à dire à les ordonner pour la relation « dépend de » (tri topologique) facile à déterminer depuis la colonne de droite. Une troisième colonne peut servir à établir ce tri, tâche qui peut être automatisée qu'on peut rapprocher de la méthode [P47] que Claude introduit à partir de 1973.

Définitions informelles	Ordre	Définitions formelles (algorithmiques)
M : Maximum de a, b, c	3	$M = \text{ si } a > mbc \text{ alors } a \text{ sinon } mbc$
mbc : maximum de b et c	2	$mbc = \text{ si } b > c \text{ alors } b \text{ sinon } c$
a, b, c : données	1	$a, b, c = \text{ lire}$

Cette méthode a été largement employée à Nancy (université, IUT, écoles, IREM et formation des professeurs du second degré), notamment pour l'initiation avec un avantage pédagogique notoire : montrer aux étudiants que la programmation, loin

de se ramener à quelques recettes, peut s'appuyer sur le raisonnement et que ce raisonnement s'enseigne.

Cependant, sa pratique ne s'est pas répandue et elle est donc restée une méthode purement nancéienne²¹. Son implantation informatique (Bernard Huc, [T34]), était complète, mais la faiblesse des techniques ergonomiques de l'époque l'ont desservie. Encore une innovation arrivée trop tôt, sans doute !

Jacques Jaray évoque cette période : « à l'IUT, avec des élèves débutants en informatique, nous avons constaté qu'en mal d'inspiration pour construire l'algorithme solution d'un problème, ils commençaient par écrire lire (données) et s'ensuivait une période de grande perplexité.

Je situe la naissance de la Méthode déductive, abrégée en MÉDÉE²², un peu après 1973. Claude était professeur au département informatique de l'IUT, il enseignait la programmation aux élèves de première année. Il avait, entre autres comme assistants : Françoise Bellegarde, Hubert Pistré, Bernard Huc et moi-même... La méthode déductive s'est éteinte au milieu des années 80, lorsque les langages objets ont été enseignés et parce qu'elle était peu naturellement adaptée pour concevoir des systèmes réactifs ».

Autres aventures : école d'été, IFIP, Algol 68, IREM...

Nous ne pouvons pas décrire toutes les facettes de l'activité de Claude. Il nous a plus d'une fois étonnés par sa manière d'accumuler les responsabilités... et de les assumer tranquillement. Nulle boulimie, mais simplement quelques facilités, une grande capacité de travail, de l'imagination, l'envie de servir et le soutien indéfectible de son épouse Monique. Évoquons ici quelques autres aspects de ses actions en lien avec la recherche.

Tout de suite après sa thèse, Claude entre en 1966 au groupe de travail WG 2.2 de l'IFIP *Formalisation of programming concepts* qui vient d'être créé. Nous avons indiqué son rôle d'animateur du groupe franco-belge sur Algol 68 (Manuel Algol 68, [P37]), groupe qui subsistera sous plusieurs formes jusqu'en 1993 avec des participants de Grenoble, Rennes, Nancy, Paris, Montpellier [16]. Du côté belge, le groupe bénéficiera notamment de la contribution déterminante de Michel Sintzoff²³, professeur à l'université libre de Bruxelles [17].

21. Mais la méthode était encore étudiée avec un oeil mathématique en 2010, bulletin N°48 de l'AP-MEP, juillet-août 2010.

22. Dénomination proposée par Pierre Lescanne et déclinée en Amédée Ducrin, nom d'auteur du livre de programmation associé à la méthode [P63].

23. C'est par Algol68 que Claude et ses élèves ont connu Michel Sintzoff lorsqu'il était chercheur au MBL (laboratoire de Philips). Michel est resté en contact avec le CRIN. Lors d'une année sabbatique passée à Nancy, il y a donné un séminaire remarquable sur les fondements de la programmation.

Jacques André : « J'avais gardé quelques contacts ténus avec le Centre de calcul, et c'est ainsi que Claude Pair m'a invité à assister à une réunion de ce qui allait devenir le Groupe Algol 68, que j'ai suivi régulièrement par la suite, et « donc » le GROPLAN de l'AF CET, ce qui m'a laissé en contact avec Pair, et le monde universitaire... ».

Le premier institut de recherche sur l'enseignement des mathématiques est apparu en 1969 à Paris. C'est en 1971 que l'IREM de Nancy est créé sous l'impulsion de Claude. Il en assure la co-direction avec Jean-Louis Ovaert²⁴.

Dès les débuts de l'informatique universitaire, il faut aussi l'enseigner, performance inédite car, en l'absence de toute formation à une matière qui n'existe pas, les étudiants recrutés comme enseignants chercheurs, ont suivi le plus souvent des études de mathématiques. Ils ignorent donc tout de cette nouvelle discipline. En 1971, Claude crée, sous l'égide de l'AF CET, une université d'été francophone²⁵ (cf. figure 7), forme originale d'auto-formation du nouveau « milieu informatique ». Elle permet d'acquérir à la fois la matière à enseigner et la pédagogie pour la présenter.

1971 Alès	1972 Neufchâtel	1973 Grenade	1974 Tarbes	1975 Rabat
1976 Lannion	1977 Montréal	1978 Namur	1979 Monastir	1980 Aix en Provence
1981 Thiès	1982 Namur	1983 Sfax	...	

FIGURE 7. École d'été d'informatique de 1971 à 1983.

C'est aussi en 1971 que sont lancés des stages de « formation approfondie » à l'informatique pour quelques dizaines de professeurs du second degré. Ces cours ont lieu sur une année scolaire dans les universités de Grenoble, Nancy, Toulouse et à l'école normale supérieure de Saint-Cloud. Claude prend en charge avec une petite équipe (Noëlle Carbonell et Brigitte Jaray) la formation de Nancy ; celle-ci s'appuie à partir de 1973 sur la méthode déductive, avec programmation en LSE²⁶. L'opération est stoppée par le ministère en 1976 et elle reprendra en 1982 à la suite d'un rapport sur l'introduction de l'informatique dans l'enseignement scolaire, confié en 1981 par le ministre Alain Savary à deux anciens présidents d'université, Yves Le Corre et... Claude Pair.

24. Jean-Louis Ovaert était un ami de Claude depuis leur scolarité au lycée Louis le Grand. Mathématicien hors pair et excellent pédagogue il a enseigné à la faculté des sciences de Nancy et plusieurs de ses étudiants ont rejoint l'équipe de Claude à la fin des années soixante.

25. Directeurs : Claude Pair de 1971 à 1974, Jean-Claude Derniame en 1975, Georges Stamon de 1976 à 1983.

26. Langage symbolique d'enseignement implémenté sur des ordinateurs Mitra 15 installés dans 58 lycées.

Conclusion

La pratique universitaire courante consiste à procéder à des évaluations au travers d'innombrables rapports établis par des comités d'experts. Ici, le lecteur l'aura compris, nous sommes dans une toute autre démarche puisque ce sont des élèves qui tentent d'évoquer les années de travail lointaines qu'ils ont vécues avec leur « maître », et grâce à lui. À l'époque, ils n'avaient pas bien conscience qu'ils vivaient une période aussi enrichissante pour eux que pour la science informatique. Mais ils avaient compris qu'ils bénéficiaient, auprès d'eux, d'une immense compétence, toujours disponible, parfois critique et avant tout visionnaire, comme ils ne l'ont réalisé que bien plus tard.

Critique, oui il pouvait être critique et parfois vivement mais toujours de manière argumentée et si habilement présentée qu'à la fin d'une réunion où nous avons tous largement divergé, nous entendions Claude conclure par sa synthèse si pertinente que nous le quittions avec le plan qu'il avait dessiné et pleinement convaincus que c'était notre plan²⁷ !

Disponible, devenu président de l'INPL (1976), Claude nous recevait encore longuement. Sur son bureau, symbole du personnage, rien d'autre qu'une feuille de papier, gratifiée de quelques lignes griffonnées²⁸ : son plan du jour.

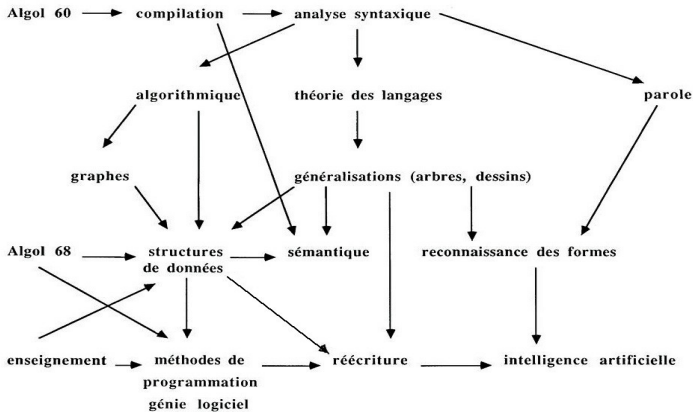


FIGURE 8. Le graphe de développement du CRIN (1963-1981) selon Claude Pair.

Encourageant, toujours à l'écoute, faisant des observations pointues, bienveillantes et constructives, et promouvant nos idées, Claude nous a incités à raisonner de façon rigoureuse et sobre.

27. Socrate parlait de maïeutique.

28. On ne peut pas dire que Claude a une belle écriture !

Fédérateur, Claude a tenu de manière constante à regrouper sans exclusion tous les enseignants chercheurs en informatique de Nancy sans tenir compte des clivages institutionnels.

Visionnaire, on peut constater que certaines publications de Claude sont encore citées plus de trente ans après leur parution²⁹.

Conclusion, c'est aussi laisser une impression synthétique au lecteur. Alors, résumons : la motivation de Claude Pair chercheur, c'était de faire de l'informatique une science. Il y a réussi. Achéons par un graphe dont il est l'auteur (cf. figure 8).

Publications de Claude Pair

- [P1] *Justification théorique de l'utilisation des piles en compilation*, CR AS 257, 3278- 3281, 1963.
- [P2] *Arbres, piles et compilation*, Revue française de traitement de l'information - Chiffres 7, n°3, 199-216, 1964.
- [P3] *Sur la détermination de la correspondance paramètre formel - paramètre effectif des procédures Algol*, congrès AFIRO, 157-162, 1964.
- [P4] *Essai de description de la sémantique des langages de programmation*, séminaire AFIRO et rapport interne, 1964.
- [P5] *Description d'un compilateur Algol*, European Région 1620 Users Group, Mannheim, 1965.
- [P6] *Étude d'un algorithme d'analyse syntaxique*, manuscrit consultable aux Archives Henri-Poincaré, <https://poincare.univ-lorraine.fr/>, Nancy, 1965.
- [P7] *Étude de la notion de pile, application à l'analyse syntaxique*, thèse d'État, Nancy, 1965.
- [P8] *Analyse syntaxique des langages de Chomsky*, exposé à un séminaire à Grenoble, 1966.
- [P9] *Sur des algorithmes pour les problèmes de cheminement dans les graphes*, Théorie des graphes, Journées internationales d'étude, Rome, Dunod - Gordon and Breach, 271-300, 1966.
- [P10] *La formalisation des grammaires*, Centre de Recherches et d'Applications Linguistiques, Faculté des lettres et sciences humaines, Nancy, 1967.
- [P11] *Définition et étude des bilangages réguliers*, avec A. Quéré, Inf. and Control 13, 565-593, mai 1968.
- [P12] *Introduction à Algol 68*, RIRO Informatique 3, n°3, 17-52, 1969.
- [P13] *Algol 60*, Techniques de l'Ingénieur H 2160, 1970.
- [P14] *Sur des notions algébriques liées à l'analyse syntaxique*, RIRO Math.4 n° 3, 3-29, 1970.
- [P15] *Sur les fonctions primitives récursives de ramifications*, avec A. Quéré, Acta Math. Acad. Sc. Hung. 21, 437- 444, 1970.
- [P16] *Mille et un algorithmes pour les problèmes de cheminement dans les graphes*, RIRO Informatique 4 n° 3, 125-143, 1970.
- [P17] *Survol de la théorie des langages*, journées AFCET, Grenoble, 1970.
- [P18] *FACE, langage pour l'écriture des compilateurs*, avec F. Bellegarde et J. Maroldt, congrès AFCET, broch. 2, 5-21, 1970.

29. Par exemple, selon Google Scholar, pour l'article sur les bilangages paru en 1968 [P11], on trouve encore 37 citations postérieures à l'an 2000. Le lecteur pourra aussi lire l'introduction prémonitoire du livre de Livercy [26].

- [P19] *Concerning the syntax of Algol 68*, Algol Bulletin 31, 16-27, 1970.
- [P20] *Rapport d'évaluation Algol 68*, avec coauteurs, RIRO Informatique 5, n° 1, 15-106, 1971.
- [P21] *Application de la théorie des ramifications au problème de l'équivalence structurale de deux C-grammaires*, RIRO Math. 5, n° 2, 130-136, 1971.
- [P22] *L'algorithme d'analyse syntaxique de P. Broize*, RIRO Informatique, n° 2, 111-123, 1971.
- [P23] *Problèmes de cheminement dans les graphes*, avec J.C. Derniame, Monographies d'informatique, Dunod, 1971.
- [P24] *Les structures de données et leur représentation en mémoire*, cours à l'Ecole d'Eté de l'AF CET, 1971, puis, avec Marie-Claude Gaudel, IRIA Roquencourt, 270 p., 1977.
- [P25] *La formalisation des langages de programmation*, Informatique et Sciences Humaines 9, n° 34, 71-86, 1971.
- [P26] *Définition du langage algorithmique Algol 68*, avec le groupe Algol de l'AF CET, Act. Sc. et Ind. 1354, Hermann, 1972.
- [P27] *Compilation*, cours à l'école d'été de l'AF CET, 1972.
- [P28] *Analyse syntaxique*, cours à l'école d'été CEA-EDF-IRIA, 1973.
- [P29] *On description languages for algorithms and programs*, rapport interne, 1973.
- [P30] *Formalization of the notions of data, information and information structure*, in IFIP Working Conference Data Base Management Systems, Klimbie-Koffemann éd., North-Holland 149-168, <https://dblp.uni-trier.de/db/conf/ds/dbm74.html#Pair74>, et 1974³⁰
- [P31] *Toute grammaire LL(k) est LR(k)*, RAIRO Informatique 8 n° 2, 59-62, 1974.
- [P32] *Aspects de la théorie de la programmation*, avec coauteurs, cours à l'école d'été de l'AF CET, 1974.
- [P33] *Rapport de l'ATP Informatique 1972 : Informatique théorique, programmes et données, Équipe associée 364*, Université de Nancy II. Éd, 1974.
- [P34] *Réflexions sur la programmation*, journée SESORI sur la programmation, 13-18, 1974.
- [P35] *Calculs, ensembles de calculs, équivalence de programmes*, Conv. Informatica Teorica 1973, Symposia Matematica 15, Academic Press, 35-54, 1975.
- [P36] *Introduction à une méthode de programmation déductive*, avec J. Maroldt in L'enseignement de la programmation, IRIA, 1975.
- [P37] *Manuel du langage algorithmique Algol 68*, P. Bacchus, J. André, C. Pair (éds), groupe Algol de l'AF CET, Act. Sc. et Ind. 1373, Hermann), 1975.
- [P38] *Algorithms for checking consistency of attribute grammars*, in Proving and Improving programs, avec B. Lorho, IRIA Int. Coll., Arc et Senans, 29-54, 1975.
- [P39] *Some proposals for a very high level language on a variable universe*, in New Directions in Algorithmic Languages (S. Schuman, ed.), W.G. 2.1. Conf., IRIA, 53- 70, 1975.
- [P40] *Du problème à sa solution*, Colloque SESORI, Logique et Programmation, Le Bischenberg, 1975.
- [P41] *Initiation à la programmation*, avec N. Carbonnel, CNTE, 1976.
- [P42] *Les arbres en théorie des langages*, Colloque « Les arbres en algèbre et en programmation », Lille, 196-216, 1976.
- [P43] *Inference for regular bilanguages*, in Formal Languages and programming (R. Aguilar, éd.), North-Holland, 15-30, 1976.

30. On peut aussi consulter la page dédiée à Claude Pair sur le site du *Digital Bibliography & Library Project*, <https://dblp.uni-trier.de/pid/63/4413.html>.

- [P44] *Objectifs, réalisations et expériences de l'introduction de l'informatique dans l'enseignement secondaire en France*, Colloque sur Informatique et Enseignement, Neuchâtel, 1977.
- [P45] *La construction des programmes*, Journées informatiques de Nice et Ecole d'été de Montréal, 1977.
- [P46] *Inference for regular bilanguages*, avec J. Berger, Journ. Comp. System Sciences 16, 100-122, 1978.
- [P47] *MEDEE, a type of language for constructing programs*, avec coauteurs, Workshop on reliable software, Bonn 1978.
- [P48] *La programmation : de l'énoncé au programme*, conférence invitée au congrès de l' AFCET, 1978.
- [P49] *Correctnes proofs of syntax-directed processing description by attributes*, avec M. Amirchahy et D. Néel Journ. Comp. System Sciences 18 , 1-17, 1979.
- [P50] *La construction des programmes*, RAIRO Informatique 13, 113-137, 1979.
- [P51] *Construction de compilateurs basée sur une sémantique formelle*, avec M.C. Gaudel, Journées francophones, Genève, 83-101, 1979.
- [P52] *The use of formal semantics to produce and prove compilers*, avec M.C. Gaudel, Workshop on semantics of programming languages, Bad Hohnef, 1979.
- [P53] *Spécifications et langages de spécification*, Panorama des langages d'aujourd'hui, séminaire GROPLAN, Congrès, 1979.
- [P54] *Computer Science, a new dimension of contemporary science*, avec M. Borillo in Scientific Culture in the Contemporary World, Scientia, 343-363, 1979.
- [P55] *L'algorithmique, ou comment l'informatique amène à faire des mathématiques*, Conférence invitée au 4e séminaire, coll. de la Comm. Int. Enst Math., Luxembourg, 117-134, 1979.
- [P56] *Some theoretical aspects of program construction*, in Program Construction, International Summer School, Springer Lecture Notes in Comp. Sc. 69, 617-651, 1979.
- [P57] *La production assistée du logiciel, introduction et perspectives*, Journées Francophones, Genève, 1-13, 1980.
- [P58] *Types abstraits et sémantique algébrique des langages de programmation*, CRIN 80-R-011, 1980.
- [P59] *Application of abstract data types to the définition of the semantics of programming languages*, conférence invitée à « Formalization of Programming Concepts », Peniscola, 1981.
- [P60] *Claude Pair et Yves Le Corre, L'introduction de l'informatique dans l'éducation nationale*, Rapport au Ministre, http://www.epi.asso.fr/revue/histo/h81_Pair-Le-Corre.htm, octobre 1981.
- [P61] *Abstract data types and algebraic semantics of programming languages*, Theor. Comp. Sc.18, 1 – 31, 1982.
- [P62] *A systematic study of models of abstract data types*, avec M. Broy et M. Wirsing Theor comp. Sc.33, 139-174, 1984.
- [P63] *Programmation*, par Amédée Ducrin (ouvrage collectif : M. Créhange, J.-P. Finance, J. Guyard, N. Hertschuh, C. Pair, M. Quéré, J. Souquières), tomes 1 et 2, Dunod Informatique, 1984.
- [P64] *Informatique et enseignement : hier, aujourd'hui et demain*, Bulletin de l'association Enseignement public et Informatique, septembre 1987.
- [P65] *Informatique et lutte contre l'échec scolaire*, in J.-M. Hoc, P. Mendelsohn, Les langages informatiques dans l'enseignement, Psychologie française 32, 1987.
- [P66] *Construire les algorithmes*, avec R. Mohr et R. Schott, Dunod Informatique, 1988.
- [P67] *Can computer help combat school failure?*, C. Pair et coll., Computers in education, Elsevier, 1988.
- [P68] *Programmation, langages et méthodes de programmation*, Le Travail humain 51, no 4, 1988.

[P69] *A tout CRIN : histoire d'un laboratoire*, Colloque sur l'Histoire de l'Informatique en France, Grenoble, 1988. in *The History of a Laboratory, Annals of the History of Computing* 12, 1990.

[P70] *Programming, programming languages and programming methods*, Psychology of Programming, Academic Press, 1990.

Thèses dirigées par Claude Pair

[T1] M. CUSEY. Construction d'un compilateur Algol pour IBM 1620, 3e cycle (1964).

[T2] R. ROMAC. Étude des méthodes de tri, 3e cycle (1964).

[T3] J. ANDRÉ. Contribution à la construction d'un compilateur Algol pour IBM 1620, 3e cycle (1965).

[T4] A. ÉMOND. Application de la notion de pile à des problèmes portant sur les chemins des graphes, 3e cycle (1965).

[T5] A. FLOC'H. Achèvement d'un compilateur Algol, traitement des procédures, 3e cycle (1966).

[T6] J.C. DERNIAME. Étude d'algorithmes pour les problèmes de cheminement dans les graphes finis, 3e cycle (1966).

[T7] H. SCHIVI. Étude de Cogent, langage de traitement de structures arborescentes, 3e cycle (1968).

[T8] A. QUÉRÉ. Étude des ramifications et des bilangages, 3e cycle (1969).

[T9] J. VILLARD. Emploi de PL/1 pour les problèmes de linguistique, 3e cycle (1969).

[T10] J.M. DIRAND. Les langages ATF-LMU, application aux problèmes linguistiques, mise en œuvre sur CII 10070, 3e cycle (1969).

[T11] J.M. MARTIN. Un mode d'exploitation du dossier médical, 3e cycle (1969).

[T12] J. CHABRIER. Etude d'une télétransmission par terminal, 3e cycle (1970).

[T13] J.M. LECLAIRE. Définition de la syntaxe des langages de programmation, 3e cycle (1970).

[T14] R. KHALIL. Essai de formalisation de la description des compilateurs, docteur-ingénieur (1970).

[T15] C. LEMAIRE. Système général d'analyse syntaxique, 3e cycle (1971).

[T16] P. LESCANNE. Étude de quelques théories des langages et généralisation du théorème de Kleene, 3e cycle (1971).

[T17] F. BELLEGARDE. Face, langage d'écriture de compilateurs, définition et implémentation, 3e cycle (1972).

[T18] J. MAROLDT. Définition de Face, langage pour l'écriture des compilateurs, implémentation d'un sous-ensemble, docteur-ingénieur (1972).

[T19] P. GERMAIN. Système de gestion et d'exploitation documentaire d'un corpus de dossiers médicaux, 3e cycle (1972).

[T20] R. MOHR. Modèle algébrique pour l'analyse syntaxique de figures, 3e cycle (1973).

[T21] N. CARBONELL. Rôle des fonctions récursives de ramifications dans la définition d'une langue naturelle, application à la syntaxe française, 3e cycle (1972).

[T22] J.B. JOUIN. Reconnaissance des fonctions primaires de la phrase anglaise, 3e cycle (1973).

[T23] J.C. DERNIAME. Le projet CIVA, un système de programmation modulaire, État (1974).

[T24] J.L. RÉMY. Structures d'information, formalisation des notions d'accès et de modification d'une donnée, 3e cycle (1974).

[T25] J.P. FINANCE. Contribution à la formalisation de la sémantique d'un langage de programmation, application à Algol 68, 3e cycle (1974).

- [T26] P. MARCHAND. Étude et classification des bigrammaires, applications à l'étude des systèmes transformationnels, 3e cycle (1974).
- [T27] H. PISTRÉ. L'analyse syntaxique dans FACE, langage pour l'écriture des compilateurs, 3e cycle (1975).
- [T28] J. JARAY. Le langage SNOBOL4, ses applications, son implémentation, 3e cycle (1975).
- [T29] M. CRÉHANGE. Description formelle, représentation, interrogation des informations complexes : système Pivoines, État (1975).
- [T30] A.M. RASSER. Outils d'aide à la mise au point d'un compilateur écrit dans le langage FACE, 3e cycle (1976).
- [T31] M. MAZAUD. Système d'aide à la production de traducteurs, 3e cycle (1976).
- [T32] J.J. GIRARDOT, F. MIREAUX. Réalisation d'un interprète complet du langage APL sur un mini-ordinateur, docteur-ingénieur (1978).
- [T33] F. PRUSKER. Aspects théoriques et pratiques du tri par fusion sur disque, État (1977).
- [T34] B. HUC. Mise en œuvre de la méthode de programmation déductive, docteur- ingénieur (1977).
- [T35] J. BERGER. A study of inference for regular bilanguages, PH.D, University of Pensylvania (1977).
- [T36] A. TISSERANT. Compilateur du langage Pascal sur mini-ordinateurs, réalisation sur Solar 16, docteur-ingénieur (1977).
- [T37] A. COCHET-MUCHY. La production de programmes dans le projet SYGARE, docteur-ingénieur (1978).
- [T38] P. NONN. Le système d'exploitation dans le projet SYGARE, docteur-ingénieur (1978).
- [T39] P. LESCANNE. Étude algébrique et relationnelle des types abstraits et de leurs représentations, État (1979).
- [T40] J.P. FINANCE. Étude de la construction des programmes : méthodes et langages de spécification et de réalisation de problèmes, État (1979).
- [T41] J. MARIN-NAVARRO. Un método de programacion .- presentacion, implementacion, transporte, Universidad Complutense, Madrid (1979).
- [T42] M. QUÉRÉ. Contribution à l'amélioration des processus d'enseignement, d'apprentissage et d'organisation de l'éducation : l'ordinateur outil et objet d'enseignement, application au projet SATIRE, État (1980).
- [T43] M.C. GAUDEL. Génération et preuve de compilateurs basées sur une sémantique formelle des langages de programmation, État (1980).
- [T44] P. DESCHAMP. Production de compilateurs à partir d'une description sémantique des langages de programmation : le système PERLUETTE, docteur-ingénieur (1980).
- [T45] P. MARCHAND. Langages d'arbres, langages dans les algèbres libres, État (1981).
- [T46] E. CHOURAQUI. Contribution à l'étude théorique de la représentation des connaissances. Le système symbolique Arches, État (1981).
- [T47] J.L. RÉMY. Etude des systèmes de réécriture conditionnels et applications aux types abstraits algébriques, État (1982).
- [T48] Radhia COUSOT. Fondements des méthodes de preuve d'invariance et de fatalité de programmes parallèles, État (1985).

Références

- [1] A. V. Aho and J. D. Ullman, *The Theory of Parsing, Translation, and Compiling*, Vol. 1, Parsing. Prentice Hall, 1972.
- [2] R. M. Amadio, Luca Cardelli, *Subtyping Recursive Types*, ACM Trans. Program. Lang. Syst. 15(4), 575-631, DOI :10.1145/155183.155231, 1993.
- [3] J.-W. Backus et coll³¹, *Report on the algorithmic language Algol 60*, *Numerische Mathematik* 2, 1960.
- [4] J. C. Boussard, *Étude et réalisation d'un compilateur Algol 60 sur calculatrice du type IBM 7090/7094 et 7040/44*, Thèse d'État Science appliquée, Grenoble, <https://hal.archives-ouvertes.fr/tel-00009449/>, 1964.
- [5] J. C. Boussard, J. J. Duby (éds), *Rapport d'évaluation Algol 68*, *Revue française d'informatique et de recherche opérationnelle R-1*, 1971.
- [6] J. Buffet, P. Arnal, A. Quéré *Définition du langage algorithmique Algol 68*, présent. et trad. française du Report on the algorithmic language Algol 68 éd. Hermann (Actualités scientifiques et industrielles), 1972.
- [7] N. Chomsky, *On certain properties of grammars*, *Information and Control* 2, 1959.
- [8] M. Créhange, *Structure du code de programmation*, Thèse 3e cycle, Nancy, 1961³². Et *Code de programmation*, Cahier n° 1 du groupement des utilisateurs scientifiques des ordinateurs IBM 650, octobre 1960.
- [9] M. Créhange, M.-C. Haton, *L'informatique universitaire à Nancy : un demi-siècle de développement*, revue de la SIF 1024 no 3, pp. 59-74. 2014.
- [10] O. J. Dahl, E.W. Dijkstra, C. A. H. Hoare, *Structured Programming*, Acad. Press, 1972.
- [11] J.C. Derniame, *A propos du cheminement dans les graphes*, revue de la SIF 1024, N° 16, 2020.
- [12] E. W. Dijkstra, *Go To Statement Considered Harmful*, CACM, Communications of the ACM 11, 3. 147-148, 1968.
- [13] J.-P. Finance, J.-L. Rémy, *Structure d'information et sémantique d'un langage de programmation*, école d'été de l'AFCEC GRENADE 1973.
- [14] J.-P. Finance, P. Lescanne, P. Marchand, R. Mohr, C. Pair, A. Quéré, J.-L. Rémy, *Aspect de la Théorie de la Programmation*, Cours de l'école d'été d'informatique, Tarbes, (imprimé à l'IREM de Nancy), 1974.
- [15] L. Fossier, M. Créhange, *Un essai de traitement sur ordinateur des documents diplomatiques du Moyen Age*, ANNALES Economies, Sociétés, Civilisations) 1, 1970.
- [16] A. Gerbier, *Mes premières Constructions de programmes*, LNCS 55, Springer- Verlag, 1977.
- [17] Anna Gram, *Raisonnement pour programmer*, Dunod informatique, 1993.
- [18] Y. Gurevich, *Evolving algebras, Lipari guide. Specification and validation methods*, 9-36, <https://dblp.uni-trier.de/db/conf/asm/eap1993.html#Gurevich93>, 1993.
- [19] Y. Gurevich : *Sequential abstract-state machines capture sequential algorithms*. ACM Trans. Comput. Log. 1(1) : 77-111, <https://dblp.uni-trier.de/db/journals/tocl/tocl1.html#Gurevich00>, 2000.

31. Le comité de définition comprend un français, B. Vauquois, professeur à Grenoble.

32. Dans l'article du journal du CNRS du 23.09.2016 par Martin Koppe, Pierre Mounier-Kuhn signale qu'il s'agit « de la première thèse de France en informatique ».

- [20] S. C. Johnson, Yacc : Yet Another Compiler-Compiler, AT&T Bell Laboratories Technical Reports. AT&T Bell Laboratories Murray Hill, New Jersey 07974 (32), <http://dinosaur.compilertools.net/yacc/>, 1975, Retrieved 31 October 2014.
- [21] D. E. Knuth. A proposal for input-output conventions in Algol 60. Commun. ACM 7, 5, 273-283, May 1964.
- [22] D. E. Knuth, *Literate Programming*, Californie, Stanford University Center for the Study of Language and Information, (ISBN 978-0937073803), 1992.
- [23] D. E. Knuth, *The Art of Computer Programming*, Addison-Wesley, <https://www-cs-faculty.stanford.edu/%7Eknuth/taocp.html>, à partir de 1968.
- [24] E. Lazard, P. Mounier-Kuhn, *Histoire illustrée de l'informatique*, 2e édition, EDP Sciences, 2019.
- [25] P. Lescanne, « *La science informatique* », dans Encyclopédie Illustrée de la Lorraine, Histoire des Sciences et des Techniques, vol. Sciences exactes, Éditions Serpenoises, p. 105-116, 1996.
- [26] C. Liverycy (nom collectif : J.-P. Finance, M. Grandbastien, P. Lescanne, P. Marchand, R. Mohr, A. Quéré, J.-L. Rémy), *Théorie des programmes : schémas, preuves, sémantique*, Dunod Informatique 1978³³.
- [27] R. Mohr, *Généralisation de la notion de langage à contexte libre. Application à l'analyse syntaxique de figures*, RAIRO Informatique théorique, tome 9, no R2, p. 55-88, 1975.
- [28] P. Naur.(éd.), *Revised Report on the algorithmic language Algol 60*, Comm. ACM, 1963.
- [29] R. Péter, *Recursive Functions*. Traduit par István Földes. New York : Academic Press, 1967³⁴.
- [30] R. Péter, *Die Pair-schen freien Binoïden als Spezialfälle des angeordneten freien holomorphen Mengens*, Eötvös Lorand Universität Budapest, 1968, publié dans Acta Mathematica Academiae Scientiarum Hungaricae Tomus 21 (3—4), pp. 29--313, 1970.
- [31] B. C. Pierce, *Types and Programming Languages*, The MIT Press Cambridge, Massachusetts, ISBN 0-262-16209-1, 2002.
- [32] M. Quéré, *Modèle mathématique d'un système d'enseignement*, Computers in education, North Holland, 1975.
- [33] D. Sangiorgi, *On the origins of bisimulation and coinduction*³⁵, ACM Trans. Program. Lang. Syst. 31(4), 15 :1-15 :41, <https://dblp.uni-trier.de/db/journals/toplas/toplas31.html#Sangiorgi09>, 2009.
- [34] J-D Warnier, B. M. Flanagan, *Entraînement à la construction des programmes d'informatique : Principes et exercices pratiques*, Les Éditions d'organisation, 1970.
- [35] A. Van Wijngaarden, B. J. Mailloux, J. E. L. Peck, C. H. A. Koster (éds), *Report on the Algorithmic Language Algol 68*, Springer, 1969 et Traduction française par le groupe franco-belge Algol P. Arnal, J. Buffet, A. Quéré, (éds.), Actualités scientifiques et industrielles, Hermann, 1972.
- [36] N. Wirth, Helmut Weber, *EULER : a generalization of Algol and its formal definition*, Part 1. Commun. ACM 9(1), 13-25 (1966) et Part II. Commun. ACM 9(2), 89-99, 1966.

33. http://denif.ens-lyon.fr/data/programmation_enslyon/2007_sem2/biblio/Liverycy.pdf.

34. Édition originale en 1951.

35. Une intéressante étude historique qui ignore cependant la contribution de Claude Pair.



Concours de programmes AFCET — 1981

Travaux des lauréats présentés par l'EPI

Jacques Baudé¹

À l'occasion de sa troisième conférence mondiale sur « Informatique et éducation » (*World conference on computer education, WCCE*) qui s'est tenue à Lausanne du 27 au 31 juillet 1981, l'IFIP (Fédération internationale pour le traitement de l'information) a demandé à chacun de ses trente-huit pays membres d'organiser un concours sur le thème « L'informatique à l'école », ouvert aux jeunes d'âge scolaire.

En France, c'est l'AFCET², en liaison avec le comité français pour la conférence de Lausanne, qui s'est chargée d'organiser ce concours national.

L'AFCET s'est naturellement tournée vers l'EPI, avec sa connaissance du terrain, pour assurer la liaison entre le comité technique, les candidats et le jury. L'année suivante, elle a mis à disposition de l'association l'ensemble des dossiers des lauréats pour la publication d'un numéro spécial (supplément au Bulletin trimestriel de juin 1982) qui a rendu compte des dossiers des seize premiers lauréats.



1. Secrétaire général puis président de l'EPI de 1981 à 1995, président d'honneur de l'EPI, membre d'honneur de la SiF, jacquesbaude@free.fr.

2. Association française pour la cybernétique économique et technique — société savante d'informatique, d'automatique et de recherche opérationnelle, qui a existé de 1968 à 1998.

Dans le cadre de cet article j'ai retenu 7 dossiers qui représentent :

- le premier et le second cycle ;
- la diversité des thèmes choisis par les élèves : simulation, traitement de données, exercices, utilitaires ;
- la diversité des matériels (Mitra15, T1600, X1, HP) et des langages (LSE, Basic, langage machine).

La décennie 70 a montré que l'informatique est à la portée des élèves qui, non seulement utilisent avec intérêt des programmes élaborés pour la plupart d'entre eux par les enseignants, mais sont capables de concevoir et de mettre au point leurs propres applications.

Un travail en profondeur a eu lieu notamment dans les clubs informatiques qui se sont ouverts dans les lycées-collèges équipés grâce, en particulier, aux enseignantes et enseignants ayant suivi une formation approfondie³ (dite formation *lourde*⁴).

C'est fort de ces constatations que le concours a été ouvert.

Les conditions de participation au concours étaient les suivantes :

- réalisation d'un programme qui représente un phénomène simple ou complexe relevant des disciplines enseignées dans le secondaire général et technique (langages de programmation autorisés : Algol, APL, Basic, Cobol, Fortran, LSE⁵, Pascal et PL/1) ;
- constitution d'un dossier comprenant une description des moyens employés, matériels et logiciels, une présentation claire et détaillée du programme, et un listing du programme et des résultats obtenus à l'exécution.

Afin de réunir, sur les dossiers des candidats, toutes les assurances techniques et les garanties d'authenticité indispensables à l'établissement d'un classement juste et objectif, le jury a tenu, dès sa première session, à étayer ses décisions sur les travaux d'un comité technique regroupant des spécialistes informaticiens et des représentants de l'administration. Ce comité s'est lui-même entouré d'experts choisis parmi les professeurs de l'enseignement secondaire à la fois spécialistes d'une discipline autre que l'informatique et informaticiens rompus aux techniques d'utilisation pédagogique de l'ordinateur.

Composition du jury :

- Président :

A. Danzin, président de l'AFCEC.

3. J. Baudé, « L'expérience des 58 lycées », bulletin de la Société informatique de France, numéro 4, octobre 2014, pp. 105–115.

4. J. Baudé, « Le séminaire de Sèvres (mars 1970) », bulletin de la Société informatique de France, numéro 11, septembre 2017, pp. 115–127.

5. J. Baudé, « Le système LSE », bulletin de la Société informatique de France, numéro 7, novembre 2015, pp. 41–56.

— Jury :

- G. Cailleteau, professeur lycée G.Fauré,
J.M. Chabanas, rédacteur en chef 01 Informatique,
A. Ducrocq, professeur Institut d'études politiques de Paris,
C. Facca, professeur CES Gustave Monod,
F. Le Lionnais, conseiller scientifique,
G. Leoutre, directeur général CNDP,
W. Mercouroff, président du comité français WCCE-81,
H. Nora, chef du service de la télématique DGT,
J.Ch. Pelissolo, membre du directoire CDF — chimie,
M. Regnard, proviseur lycée de Sèvres,
P. R. Sallebert, délégué général FIEE,
M. Saurel, directeur des lycées — ministère de l'Éducation nationale,
G. Septours, directeur INRP,
P. Siksik, professeur CES Henri Walon,
J. Treffel, directeur, adjoint au directeur général, chargé de la prospective et du développement des moyens techniques et scientifiques dans l'éducation, ministère de l'Éducation nationale.
- Membre du jury chargé des liaisons avec le comité technique :
- J. Baudé, professeur au lycée Pierre Corneille — secrétaire général de l'EPI.
- Responsables du comité technique :
- J.C. Boussard, professeur à l'université de Nice,
N. Carbonell, maître-assistant à l'université de Nancy.
- Secrétaire du jury et du comité technique :
- J. Lancereaux, AFCET.

Quelques informations sur l'ensemble des candidats et leurs dossiers

Cent douze dossiers complets sont parvenus à l'AFCET dans le délai imparti. Neuf d'entre eux ont été écartés car ils ne répondaient pas aux normes établies par le règlement du concours : programmation exclusive en langage machine, choix d'un sujet trop éloigné du thème du concours, ou proximité excessive de résultats connus. Tous les dossiers retenus, soit 103, ont fait l'objet d'une analyse minutieuse de la part

des professeurs experts de la discipline considérée. Ces derniers ont été unanimes à reconnaître l'importance du travail fourni par chaque candidat et la maturité d'esprit dont tous ont fait preuve ; une large proportion des dossiers présente en outre, selon eux, un réel intérêt pédagogique ; ces observations constituent sans aucun doute la conclusion la plus riche de promesses à porter au crédit du concours. C'est donc parmi un nombre important de réalisations, toutes dignes d'intérêt, que le jury a eu la lourde tâche de distinguer les candidats méritant le plus, à ses yeux, de figurer au palmarès. Une étude statistique des dossiers retenus fournit les informations suivantes (extrait) :

— âge des candidats :

Année de naissance	1962	1963	1964	1965	1966	1967	1968
Nombre de candidats	19	32	34	8	5	4	1

— genre des candidats : 12 filles, 91 garçons ;

— classe des candidats :

1 ^{er} cycle	6 ^e	5 ^e	4 ^e	3 ^e
Nombre	1	2	9	4

2 ^e cycle	2 ^{de} C	2 ^{des} T1 T2 T3	1 ^{re} C	1 ^{re} D	1 ^{re} E	1 ^{re} G2
Nombre	26	10	23	4	5	3

2 ^e cycle	1 ^{re} H	Trm. B	Trm. C	Trm. D	Trm. E	Trm. F1
Nombre	1	1	8	1	1	1

Autres	CPGE	DUT Informatique 1 ^{re} année
Nombre	2	1

— discipline concernée par les dossiers :

Discipline Dossiers	Dessin 1	Hist.-Géo. 7	Musique 2	Lettres-Langues 16	Math. 36
Discipline Dossiers	Gestion 5	Tech. Indus. 2	Sc. Nat. 9	Phys.-Chimie 24	Autres 1

45 dossiers sont utilisables dans le premier cycle, 73 dans le second. Les programmes qui s'adressent indifféremment aux deux cycles ont été comptés deux fois.

Les conditions de réalisation sont les suivantes :

- 80 dossiers sont des œuvres individuelles, 23 résultent d'un travail de groupe le plus souvent de 2 membres ;
- support matériel :
 - Mitra 15 (31 dossiers),
 - T1600 (28),
 - Apple (4),
 - Commodore (8),

- HP (5),
 - LX500 (4),
 - TRS-80 (11),
 - X1 (6);
- ne sont pas mentionnés les matériels utilisés par un seul candidat ;
- support logiciel :
 - LSE⁵ (60),
 - Basic (41),
 - Cobol (1),
 - Fortran (1);
 - environnement et points d'appui informatiques :
 - un des 58 lycées⁶ équipés de mini-ordinateurs (MITRA15 ou T1600) : 65 dossiers,
 - un des lycées équipés de micro-ordinateurs : 11 dossiers,
 - établissements équipés de leur propre initiative : LEG (10), LEP (2), LET (3), collège (6), autres (6),
 - 5 candidats seulement ont cherché un soutien informatique en dehors des établissements scolaires.

Les lauréats titulaires des 16 prix ont utilisé LSE(10) et Basic(6) comme langage de programmation avec le matériel suivant :

- mini-ordinateurs du lycée-collège : MITRA15 (6), T1600 (4);
- micro-ordinateurs (dans l'établissement) : X1 (3), LX500 (1);
- micro-ordinateurs (à l'extérieur) : HP9830 (1), PET (1).

Les 16 dossiers primés relèvent de disciplines variées avec une nette prédominance quantitative des disciplines scientifiques : mathématiques, physique-chimie, sciences naturelles recueillent chacune quatre prix. Il est important de noter également que de nombreux dossiers présentent une orientation pluridisciplinaire.

Une première constatation s'impose : les auteurs des programmes primés se sont tous efforcés de transmettre simultanément à l'utilisateur un savoir-faire ou des connaissances, présentés de manière attrayante, et une démarche intellectuelle et un mode de pensée. On trouve donc :

- des exercices d'acquisition, de révision ou de contrôle dominant (7 dossiers); cela correspond certainement à une préoccupation importante des élèves, d'autant plus forte qu'ils sont plus jeunes (cf. les prix des collèges);
- des programmes de simulation, au nombre de six, qui permettent d'appréhender des phénomènes complexes qu'il est difficile, sinon impossible d'étudier expérimentalement dans le cadre de l'enseignement secondaire ; plusieurs candidats ont réalisé un travail de modélisation tout à fait intéressant, faisant

6. Les 58 lycées (rubrique historique de l'EPI), <https://www.epi.asso.fr/revue/histo/h70-58lycees.htm>.

ainsi la preuve de leur aptitude à maîtriser le phénomène qu'ils avaient choisi d'étudier ;

— deux dossiers mettent l'accent sur une autre utilisation privilégiée de l'ordinateur : le traitement de données (avec, en particulier, la mise en œuvre d'outils statistiques complexes et l'analyse de volumes importants d'information) ;

— un dossier, que l'on peut classer dans les programmes utilitaires (procédures graphiques), assure le lien avec les disciplines artistiques (analyse et composition de dessins géométriques).

Sept dossiers parmi les 16 lauréats et les 103 dossiers retenus

J'ai extrait de ces dossiers très complets les informations qui rendent compte du travail mais aussi des motivations et de l'enthousiasme de ces jeunes élèves. Le plus souvent je cite les lauréats tels qu'ils s'expriment.

La physiologie au service du sport, simulation d'une course à pied

Auteur : Éric Muraine

Établissement : lycée Montaigne — Bordeaux

Matériel utilisé : X1 (société Occitane)

Langage : Basic

Programme : 120 lignes

Niveau : second cycle

Prix : 1^{er} prix AFCET — IFIP

Eric Muraine : « *Mes deux hobbies sont l'informatique et les sciences naturelles. Dans ce programme, j'ai pu créer un lien entre eux et par là-même faire d'un phénomène qui me semblait irrationnel, une réalisation mathématique et donc rationnelle. Mon travail a consisté en :*

- *recherche et documentation sur les mécanismes musculaires et respiratoires ;*
- *recherche des facteurs limitant une performance (l'absorption maximale d'oxygène, dette, rythme cardiaque, etc.) ;*
- *recherche des conditions nécessaires à l'établissement d'une performance ;*
- *établissement d'un programme avec simplification des entrées (on aurait pu demander la surface corporelle, le volume maximal d'O₂ et d'autres paramètres, on ne demande plus que la taille, le poids, etc.) ;*
- *tentative de faire un programme « complet » en rajoutant le tableau de la surface corporelle ainsi que la dépense de calories ;*
- *test du programme, en comparant les valeurs réelles et les valeurs théoriques.*

D'un point de vue limité, simplement pratique, ce programme constitue un test beaucoup trop compliqué, car il est considérablement plus simple de chronométrer directement l'homme pendant qu'il est en train de courir ou de nager. Mais du point de vue pédagogique, c'est un test qui fait impression parce qu'il montre que l'aptitude de l'homme à battre des records obéit à des équations mathématiques. L'intérêt du programme est aussi en tant que potentialité : il permet à un sportif, ou à n'importe qui, de prendre connaissance de ses capacités propres, de ses espérances, de ses limites à partir des paramètres de mensurations. »

Électrolyses à anode soluble

Auteur : Corinne Margotat (en classe de 4^e)

Établissement : collège-lycée Banville à Moulins

Matériel utilisé : Mitra 15

Langage : LSE

Programme : 112 lignes

Niveau : 1^{er} cycle

Prix : 1^{er} prix des collègues

Corinne Margotat : « *Ce travail constitue une vérification des connaissances acquises après la leçon sur les électrolyses à anode soluble. Il vérifie la connaissance du vocabulaire : sens du courant, électrode. Commentaire suivant le nombre de réponses justes. Il propose différentes expériences successivement et pose des questions pour chacune d'elles. Un commentaire adapté est prévu pour chaque réponse fautive. Commentaire final sur les bonnes connaissances de ces notions.*

C'est donc un programme qui vérifie si on a appris et compris notre leçon. Des professeurs peuvent s'en servir. Je crois qu'il faut travailler en s'amusant ce qui n'est pas toujours le cas. »

Recherche des pronoms dans un texte

Auteur : Éric Le Bras (14 ans)

Établissement : collège-lycée Victor Duruy — Paris

Matériel utilisé : T1600

Langage : LSE

Programme : 147 lignes

Niveau : 1^{er} cycle

Prix : 2^e prix du 1^{er} cycle

Éric Le Bras : « *L'idée de ce programme m'est venue d'un exercice demandé en classe de 5^e du lycée Victor Duruy et que personne n'avait réussi. On s'est aperçu qu'on oubliait toujours l'une ou l'autre catégorie et que finalement les pronoms*

constituaient une part très importante du texte. Peut être utilisé en classe ou en travaux dirigés. Une originalité est aussi le fait que l'on tutoie l'élève, rendant ainsi le programme plus vivant, ainsi que le contact avec l'ordinateur.

Le programme affiche un texte contenu dans un fichier. Puis l'utilisateur doit faire une recherche des pronoms contenus dans ce texte. Correction et affichage de la note sur vingt.

L'utilisation du programme est relativement simple pour des élèves débutant en informatique. Néanmoins, la présence du professeur est indispensable pour expliquer les lacunes de chacun.

Certains élèves qui ont utilisé le programme (notamment un élève de terminale), montraient une méconnaissance totale ou partielle des pronoms.

Ce programme n'est pas seulement une interrogation, il est aussi un moyen de révision inhabituel, grâce à ses banques de données consultables. Après quelques modifications, il pourra traiter différents textes. »

Approche anthropologique d'une population hétérogène à travers l'étude de ses données hématologiques

Auteur : Pierre Bayourthe

Établissement : lycée Saint Sernin — Toulouse

Matériel utilisé : T1600

Langage : LSE

Programme : 120 lignes

Niveau : second cycle

Prix : prix spécial du jury

Pierre Bayourthe : « *Ce programme fait appel à trois disciplines : l'informatique, la biologie (génétique) et les mathématiques (statistiques) mais les données traitées sont essentiellement biologiques. Il se propose en effet d'interpréter des résultats relatifs à la fréquence de certains gènes dans une population hétérogène pour analyser les liens entre les différents groupes humains qui la composent.*

Mon premier travail a consisté à réunir une bibliographie importante tant biologique qu'informatique. C'est ainsi que je suis entré en contact avec le centre d'hématologie du CHU de Purpan qui, grâce aux conseils des chercheurs, m'a permis d'entamer la conception globale des modules de programmes, les adaptant aux données à traiter.

Je suis passé ensuite à l'écriture proprement dite des programmes et aux premières exécutions d'essais. Puis, lorsque la partie programmation a été terminée, je suis passé à la présentation des sorties, aux commentaires informatiques et à la rédaction du dossier. Enfin, j'ai réalisé les exécutions sur données réelles avant de les faire interpréter au centre d'hématologie et d'envoyer le dossier.

Ce travail m'a permis d'approfondir mes connaissances en ces matières, d'avoir une ouverture sur les milieux scientifiques et leurs méthodes particulières de travail. Il présente donc un prolongement dans le milieu scolaire de travaux scientifiques en les rendant plus facilement abordables. »

Simulation d'une réaction en chaîne entre le chlore et l'hydrogène (application à l'étude d'une vitesse de réaction à rendement quantique)

Auteurs : Gérard Tremblay, Michel Jorillon, Franck Herault

Établissement : lycée Guy Chauvet — Loudun

Points d'appui : lycée Rabelais à Chinon et IREM de Poitiers

Matériel utilisé : HP 9830 et table traçante HP 9842

Langage : Basic

Programme : 120 lignes

Niveau : second cycle

Prix : 2^e prix

Gérard Tremblay, Michel Jorillon et Franck Herault : « *On simule, sur une grille de dimensions données, les chocs successifs entre des radicaux chimiques et des molécules de chlore et d'hydrogène. On s'intéresse à l'évolution du nombre de molécules de chlorure d'hydrogène formées en fonction du nombre de chocs jusqu'à une phase d'arrêt par réaction entre deux radicaux.*

Cette simulation présente, de manière différente de l'exposé magistral classique, le mécanisme complexe d'une réaction en chaîne.

Les résultats présentés sont seulement partiels car notre temps d'utilisation du matériel a été strictement limité à quelques heures (deux mercredi après-midi) à 60 kilomètres de notre lycée. »

Géométrie analytique et ordinateur (analyse et composition de dessins géométriques)

Auteur : Éric Chantrelle

Établissement : lycée polyvalent rive gauche — Toulouse

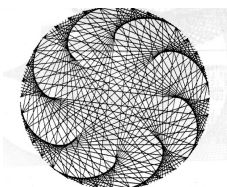
Matériel utilisé : Mitra 15, console graphique Tektronix 4006

Langage : LSE

Programme : 295 lignes

Niveau : second cycle

Prix : 3^e prix *ex-æquo*



Éric Chantrelle : « *L'ordinateur présente à l'utilisateur trois méthodes de dessin sur une console graphique. Pour chacune d'elles, le principe est expliqué à l'aide de schémas et fait appel aux connaissances de géométrie analytique enseignées dans le secondaire. Lorsque l'utilisateur a compris le procédé employé, on lui propose de réaliser ses propres dessins, après avoir donné des exemples prédéterminés ou*

choisis aléatoirement. Aidé par l'ordinateur qui lui indique certaines variables suffisantes pour déterminer les dessins, il peut alors en composer, en changer les paramètres à son gré et étudier ainsi les variations apportées aux dessins.

L'originalité de ce programme vient du fait que la discipline mathématique dont il s'agit est abordée par l'intermédiaire du dessin. La composition même de l'écran de la console graphique (matrice de points) et les modes de tracé ne sont abordables que par une étude analytique. L'utilisation d'une telle console permet de mettre en pratique les principes de géométrie acquis en cours. Il existe certains programmes de mathématiques pures sur cette console (tracé de vecteurs, de droites, études de fonctions). L'intérêt de ce programme est qu'il apporte à la géométrie une réalisation matérielle qui échappe à l'univers strict des mathématiques. »

Lecture efficace (élargissement du champ de vision)

Auteurs : Loïc Volant-Vettu, Philippe Daigne, Philippe Doussin, Catherine Gautier, Franck Lusteau, Loïc Rideau, Fabrice Volard

Établissement : E.T.P de La Salle — Nantes

Points d'appui : Microtel-club ETS-Nantes

Matériel utilisé : PET ou CBM 2001

Langage : Basic + langage machine

Programme : 218 lignes

Niveau : 1^{er} et 2^e cycles

Prix : 5^e prix *ex-æquo*

Les lauréats : « *Nous sommes dans l'ère de l'audio-visuel. Or, à la surprise générale, la production de textes imprimés n'a pas diminué face à la concurrence redoutable de la radio et surtout de la télévision. Au contraire, la quantité de livres imprimés chaque année croît dans tous les pays du monde. Or, on ne nous a jamais appris à bien lire, l'apprentissage de la lecture a 1 000 ans de retard.*

De nouvelles méthodes sont apparues récemment. Les nier serait nier le progrès. Elles constituent tout un apprentissage dont le premier objectif est l'élargissement du champ de vision. C'est l'objet de ce programme. La lecture efficace peut se résumer dans les trois mots suivants : vitesse — compréhension — mémoire. Le but de ce premier programme est de développer le champ de vision, c'est-à-dire d'augmenter la vitesse de lecture et donc la mémoire.

À notre connaissance, il n'existe aucun programme de ce genre, c'est donc son originalité. Nous avons voulu sortir des sentiers battus (sciences, mathématique, technique) et montrer qu'il était possible de réaliser des programmes d'un grand intérêt pédagogique dans un domaine qui, apparemment, ne s'y prête pas. Quant à l'intérêt, il suffit de constater que de grandes entreprises offrent à leurs cadres cette formation, que des universités et le CNRS l'ont inscrite dans leurs cycles de formation. Les résultats obtenus sont concluants : cette méthode permet à chacun de doubler sa vitesse de lecture, de comprendre mieux et de se souvenir davantage.

Une difficulté essentielle a été que nous voulions que l’affichage des lettres puisse atteindre au moins le centième de seconde. Or, avec le langage BASIC, on ne pouvait guère descendre au-dessous du soixantième de seconde. Il nous a fallu prévoir un petit sous-programme en langage machine, mais chargé par le BASIC, afin d’atteindre l’objectif que nous nous étions fixé.

Une autre difficulté a été celle de la continuité de l’apprentissage sous surveillance unique de l’ordinateur. L’idée de constituer un fichier sur cassette nous est donc venue afin que l’élève puisse redémarrer à la séance suivante où il en était. »

Extrait de la conclusion du numéro spécial

Je cite Jacques Baudé, François Fillol et Jean-Pierre Prévité, « Concours de programmes AFCET — 1981 », supplément au bulletin EPI de juin 1982 :

« Nous espérons que le lecteur a pu se faire une idée suffisamment précise sur les travaux récompensés par le jury et, par la même occasion, sur l’esprit dans lequel ils ont été élaborés. Nous avons dû résumer trop brièvement certains dossiers, ce dont les lauréats voudront bien nous excuser, tenus que nous étions par le cadre de ce numéro spécial. Rappelons, à cet égard, que tous les dossiers, de tous les candidats, peuvent être consultés à l’AFCET et qu’il est toujours possible de prendre contact directement avec les auteurs⁷.

Nous voudrions également souligner que compte tenu de la qualité d’ensemble des travaux soumis à son appréciation, la tâche du jury a été à la fois lourde et particulièrement délicate : notre but en publiant ces dossiers sous cette forme n’est pas de permettre l’élaboration d’un contre-palmarès, mais de faire passer au mieux une information que nous jugeons utile, et qui aurait été en grande partie perdue sans cette initiative de l’EPI.

Nous croyons pouvoir ajouter que la plupart de ces travaux n’auraient pas été menés à bien sans l’ambiance intellectuelle des clubs informatiques qui, en 1980, fonctionnaient depuis de nombreuses années dans les établissements équipés à titre expérimental de mini-ordinateurs.

Il faut saluer, à cet égard, l’enthousiasme des élèves et de leurs professeurs. Ces derniers n’ont pas compté leur temps dans la création, l’organisation et l’animation de ces clubs, allant bien au-delà des heures de décharge de service attribuées.

A nos yeux, la réussite de ce concours tient à la réussite même de ces clubs. De nombreux élèves ont trouvé (et trouvent encore) dans ce cadre matière à un engagement actif et volontaire.

De plus, d’autres rapports s’établissent avec les enseignants, avec le savoir ; une interdisciplinarité certaine s’y manifeste.

Nous déplorons que les choses aient si mal évolué depuis deux ans : le contingent d’heures de décharge par établissement a été considérablement réduit, ces

7. Précision donnée en 1982, date de publication du numéro spécial EPI.

heures sont notifiées tardivement aux intéressés et se traduisent le plus souvent en heures supplémentaires. Les enseignants, engagés parallèlement sur des projets de recherche, ne trouvent plus le temps matériel de s'occuper des clubs : beaucoup ont fermé ou réduit leur activité, faute d'encadrement. Il est à craindre que nos collègues finissent par se décourager et qu'à trop tirer sur la ficelle du bénévolat on nuise à la qualité du travail effectué.

A cet égard, nous considérons que ce numéro spécial est un témoignage : un tel concours a montré à l'évidence qu'en matière d'informatique pédagogique il est possible de placer et de maintenir la barre à un haut niveau ; la condition est que l'on inscrive le travail dans une démarche globale et finalisée. Celle-ci ne doit pas seulement prendre en compte l'équipement des établissements en matériel et la diffusion de didacticiels « tout faits ». Une telle attitude restrictive conduit à se priver d'aspects fondamentaux que nous avons eu l'occasion de mettre en évidence plus haut. »



Hommage à Philippe Aigrain

François Pellegrini¹



15 juillet 1949 — 11 juillet 2021

Il m'est très difficile de vous parler de Philippe Aigrain. Au delà de la grande tristesse que j'ai ressentie en apprenant son décès au cœur de la montagne pyrénéenne qu'il adorait (son identifiant Twitter était « @balaitous »), l'étendue des domaines qu'il a arpentés, de ses contributions théoriques et de ses actions pratiques, des communautés qu'il a irriguées voire fondées, peut donner le vertige. Que dire ?

Il était d'abord un informaticien, enthousiasmé par les possibilités d'échanges offertes par les systèmes et réseaux numériques, et des ponts que ceux-ci pourraient créer entre les personnes et les communautés. Après une thèse de doctorat soutenue à l'université Paris VII sur l'interprétation des expressions du langage APL, il orienta ses recherches vers l'analyse et l'indexation de ressources multimédia et les interfaces homme-machine, dirigeant jusqu'en 1996 une équipe de recherches à l'IRIT dans ce domaine.

Dès ses premières actions en faveur des radios libres, visant à supprimer le monopole de l'État sur les fréquences hertziennes, ont transparu deux lignes-forces de sa personnalité. La première est la nécessité de conceptualiser finement chaque sujet, en termes de jeux d'acteurs et d'enjeux économiques et sociaux. La deuxième est

1. Professeur d'informatique à l'université de Bordeaux.

l'engagement personnel dans l'action, selon l'adage « *penser global, agir local* », choisissant toujours l'angle d'attaque permettant de maximiser son efficacité.

C'est ainsi qu'il quitta le monde de la recherche académique pour aller travailler, de 1996 à 2003, à la direction générale « Société de l'information » (DG XIII, puis « InfSo ») de la Commission européenne. Il s'employa à y insuffler des politiques favorables aux logiciels libres, qu'il considérait comme les clés d'accès du plus grand nombre au monde numérique. Il y lutta contre les tentatives de mainmise des acteurs économiques dominants sur le savoir et les capacités industrielles de l'Europe dans le domaine du logiciel, en s'opposant notamment à la légalisation en Europe des sinistres « brevets logiciels ». Il participa à convaincre Michel Rocard, alors député européen, de s'engager dans cette bataille politique (ils se connaissaient bien, leurs deux pères, précurseurs de la physique nucléaire française, ayant été amis). Puis, constatant que ses propositions étaient régulièrement contrées au sein de l'institution européenne, et convaincu qu'il pourrait agir plus efficacement hors de ses murs, il en partit sans regret.

Tout au long de sa vie, Philippe Aigrain s'attacha à promouvoir les « biens communs » et, notamment, les « communs numériques ». Il considérait ceux-ci comme partie intégrante d'une « autre mondialisation », bénéfique, et devant être intégrés à l'agenda du mouvement alter-mondialiste en cours de structuration dans une « coalition des communs ». Investi dans le collectif européen *Communia*, il fut l'un des principaux contributeurs au *Manifeste pour le domaine public* (2010).

Témoin du lobbying acharné des multi-nationales (du divertissement, de la génétique, de l'agro-chimie, etc.) pour privatiser les communs, et fidèle à sa méthode, il analysa et exposa ces enjeux dans un ouvrage accessible au plus grand nombre : *Cause commune, l'information entre bien commun et propriété* (2005). C'est dans cet esprit qu'il lutta contre les lois DADVSI et Hadopi, et définît à cette occasion un mécanisme alternatif de rémunération des auteurs, la « contribution créative » (aux effets similaires à la « licence globale » proposée par d'autres mais qu'il trouvait juridiquement peu solide). Ce dispositif, devant permettre l'accès de tous au savoir numérisé et son partage à but non-lucratif contre une contribution forfaitaire, fut exposé dans un autre ouvrage : *Internet & Création* (2008), pied-de-nez au titre officiel de la loi Hadopi. Ironie de l'Histoire : le rejet de sa proposition a conduit à la situation dans laquelle les industries du divertissement passent sous les fourches caudines de plate-formes numériques qui facturent au public un accès forfaitaire, sorte de « licence locale », mais dont seulement une partie minoritaire de la valeur revient aux auteurs et ayants droit.

Sur le plan opérationnel, afin de structurer les actions citoyennes sur l'ensemble des sujets de société touchant aux libertés dans le monde numérique, et soucieux de former les jeunes générations, il co-fonda en 2008 l'association *La Quadrature du Net*, à la gouvernance de laquelle il participa jusqu'en 2017. Il créa une société,

Sopinspace, editrice d'outils numériques pour la participation collaborative (notamment à travers l'outil *Co-ment* d'annotation de propositions de textes législatifs). Il dirigea également la maison d'édition *Publie.net*, fournissant des ouvrages au format papier et numérique sans « verrous numériques » (DRM).

Pour Philippe Aigrain, l'avenir de la société passait par la solidarité entre les individus, en outrepassant si nécessaire les logiques parfois étroites des gouvernements. Indigné par la violence économique des institutions de l'Union européenne à l'encontre de la Grèce dirigée par Syriza, il co-fonda l'association *Interdemos*, pour organiser une solidarité de peuple à peuple. Par la suite, s'opposant au refoulement et à l'accueil hostile des migrants, il co-rédigea un manifeste humaniste pour une politique alternative, incarnée par le slogan : « J'accueille l'étranger ».

Informaticien, romancier, poète, éditeur, défenseur des libertés à l'ère numérique et de la dignité humaine, Philippe Aigrain incarnait pleinement l'« honnête homme » du siècle des Lumières. Il était aussi résolument moderne, tant par sa vision réticulaire du monde et de ses interactions, que par ses goûts artistiques. Il était tout cela et bien plus encore : une personne bienveillante, le compagnon de route de beaucoup, qui aura contribué à forger les convictions de plusieurs générations d'activistes et de penseurs des espaces numériques.

Merci et adishatz, Philippe.

Références

- [1] *Cause commune*, éditions Fayard, 2005, https://paigrain.debatpublic.net/?page_id=160
- [2] *Sharing, Culture and the Economy in the Internet Age*, Presses de l'Université d'Amsterdam, 2011, https://paigrain.debatpublic.net/?page_id=3968
- [3] *Sœur(s)*, roman abordant les enjeux de surveillance, éditions Publie.net, 2020, <https://www.publie.net/livre/soeurs-philippe-aigrain/>
- [4] Quelques poèmes de Philippe Aigrain, <https://www.atelierdebricolage.net/>



Hommage à Paul Camion

Anne Canteaut¹, Louis Goubin², Jacques Patarin²



Paul Camion en 2016.

Nous avons la tristesse de vous annoncer le décès de Paul Camion, cryptologue et spécialiste des codes correcteurs d'erreur. Paul Camion était né le 9 mai 1932 à Uccle en Belgique. Il a été enterré le 3 septembre 2021 à Bouillon (Belgique).

Diplômé de la Faculté polytechnique de Mons en 1954, Paul Camion devient chercheur au Centre d'études de recherche opérationnelle à Bruxelles en 1959. Il séjourne au centre de recherche d'IBM à Yorktown Heights en 1963. Il soutient une thèse d'État en 1967, à l'université libre de Bruxelles et à l'université Toulouse-III-Paul-Sabatier. Il entre ensuite au CNRS en tant que maître de recherches.

Il effectue plusieurs séjours aux États-Unis, notamment une année au centre de recherches en mathématiques de l'université du Wisconsin à Madison en 1969 et un séjour en 1973 au département de statistiques de l'université de Caroline du Nord à Chapel Hill.

Paul Camion acquiert la nationalité française en 1976. En 1977, il devient membre du projet Algo d'Inria (alors encore IRIA). Il dirige ensuite le projet Codes d'Inria

1. Directrice de recherche Inria.

2. Professeur à l'université de Versailles St. Quentin-en-Yvelines.

consacré aux codes correcteurs et à la cryptographie, projet qui se développe à partir de 1984 et connaît un développement soutenu au cours des années 1990. Il marquera aussi dans ce projet tous les étudiants qu'il a guidés pendant leur thèse (dont Anne Canteaut, Hervé Chabanne, Jacques Patarin, Nicolas Sendrier. . .)

Paul Camion, directeur de recherche au CNRS, quitte son poste à Inria en 1997. Il devient par la suite directeur de recherche émérite et membre collaborateur de l'équipe Combinatoire et optimisation de l'Institut de mathématiques de Jussieu-Paris Rive Gauche.

Paul Camion a obtenu la médaille d'argent du CNRS en 1976. Il laisse le souvenir d'un directeur de thèse exceptionnel, d'un algébriste de tout premier plan mondial, et d'une extrême gentillesse.



Hommage à Claudine Hermann

Florence Sèdes¹



Claudine Hermann lors d'un colloque à l'institut Henri-Poincaré, à Paris, le 13 novembre 2012. ©Antoine Tavenaux.

Cofondatrice de l'association « Femmes et sciences » qui a fêté ses vingt ans en 2020 et dont elle a été la première présidente, puis la présidente d'honneur, Claudine Hermann, illustre physicienne, première femme professeure de physique à l'École polytechnique en 1992, est décédée le 17 Juillet 2021.

Inlassablement engagée avec bienveillance et sollicitude pour promouvoir les femmes de sciences et les sciences auprès des jeunes filles, Claudine a rayonné au-delà de ses prestigieuses implications nationales en participant à la création de la plate-forme européenne des femmes scientifiques (EPWS) qui réunit une cinquantaine d'associations européennes et 15 000 femmes scientifiques européennes. Elle a activement participé, de 2015 à 2018, au projet européen Hypatia, qui a permis la conception d'une série d'activités pour les jeunes de 13 à 18 ans, en vue de les encourager et d'encourager tout particulièrement les filles toujours minoritaires dans certaines filières comme les maths, la physique, les sciences de l'ingénieur ou l'informatique, à choisir des filières et des métiers scientifiques.

1. Professeure d'informatique à l'université Toulouse 3 et chargée de mission Femmes et informatique à la SIF.

Toujours disponible lorsqu'il s'agissait de donner un avis, de bon conseil, prête à mettre en relation toute personne intéressée avec des membres de son immense réseau... Tout en simplicité et gentillesse, proche de toutes et de tous, et toujours humble dans l'excellence, mais pour toutes nos causes et actions, tenace et opiniâtre !

Claudine Hermann, remarquable pionnière, admirable femme d'action, aura marqué, stimulé, inspiré les actrices et les acteurs de l'égalité entre les femmes et les hommes dans le domaine des sciences et des technologies.

En un (bon) mot, avec son sourire même dans la maladie et son rire en cascade : « *Souvent les jeunes filles me demandent si je suis féministe. Si être féministe c'est vouloir que les femmes trouvent leur juste place dans la société, oui !* ».



Sophia Antipolis : le père fondateur s'en est allé

Gérard Giraudon¹



Pierre Laffite en 2006, source Wikipédia.

Rien ne prédestinait Pierre Laffitte à devenir le père fondateur de la première technopole d'Europe qui a bâti sa réputation sur l'informatique et les télécoms bien qu'il soit né le 1^{er} janvier 1925 à Saint-Paul-de-Vence, soit à quelques kilomètres de ce qui deviendra Sophia Antipolis, la cité de la sagesse ; ou peut-être est-ce d'avoir fréquenté l'école Freinet à Vence qui lui a permis de sortir des silos et d'avoir toujours un coup d'avance mais aussi d'être un homme d'action. En effet, après sa sortie de l'École polytechnique (promotion 1944), Pierre Laffitte poursuit ses études à l'École des mines de Paris et devient ingénieur géologue. Très vite, il déploie ses talents de bâtisseur et d'organisateur à travers son engagement pour créer le BRGM. Ses talents le conduisent à intégrer, en 1963, l'École des mines de Paris où il transforme la scolarité du corps des mines. Entre temps, il a publié au début des années 60 une tribune dans le journal *Le Monde* intitulé « Le Quartier Latin aux champs ». Dans cette tribune, il développe ce qu'il appelle la « fertilisation croisée ». C'est-à-dire la nécessité de briser les carcans, de rompre les barrières entre l'enseignement, la recherche publique et le monde de

tion 1944), Pierre Laffitte poursuit ses études à l'École des mines de Paris et devient ingénieur géologue. Très vite, il déploie ses talents de bâtisseur et d'organisateur à travers son engagement pour créer le BRGM. Ses talents le conduisent à intégrer, en 1963, l'École des mines de Paris où il transforme la scolarité du corps des mines. Entre temps, il a publié au début des années 60 une tribune dans le journal *Le Monde* intitulé « Le Quartier Latin aux champs ». Dans cette tribune, il développe ce qu'il appelle la « fertilisation croisée ». C'est-à-dire la nécessité de briser les carcans, de rompre les barrières entre l'enseignement, la recherche publique et le monde de

1. Directeur de recherche Inria — émérite.

l'entreprise ; faire en sorte que chacun puisse enrichir (c.a.d. fertiliser) l'autre et en retour être enrichi par lui. Qui dit fertilisation dit terreau et, par analogie, avec la définition qu'en a donné A.G. Tansley en 1935, on parle aujourd'hui de l'importance des écosystèmes, de l'innovation qui naît aux frontières, de la nécessaire diversité des approches scientifiques et économiques conditions pour aborder les problèmes complexes auxquels notre monde fait face. Cette vision, Pierre Laffitte, l'a portée dès le début des années 60 et, parce que c'était un bâtisseur, il a décidé de l'incarner et de la faire vivre sur un territoire qu'il a appelé Sophia Antipolis, « la cité internationale de la sagesse, des sciences et des techniques ». Sur ce territoire, initialement sur Valbonne puis à cheval sur cinq communes il n'y avait rien, du moins rien qui ressemble à une cité internationale ; il y avait certes des écosystèmes mais ils étaient de type méditerranéen sur un paysage de pinèdes et de garrigues traversé de routes forestières, parsemé de quelques champs d'olivier et parcouru par des sangliers. Ce fut un long combat à l'aube des années 1970 et en profitant de la politique d'aménagement du territoire de l'État portée par la DATAR, sa force de persuasion et son important réseau personnel permirent que Sophia Antipolis fut créée en 1969. Au départ, il faut bien le dire comme une « colonie parisienne » avec bien sûr l'arrivée d'une antenne de l'École des mines de Paris, puis du site informatique d'Air France sans qui Amadeus, leader mondial de son marché ne serait pas à Sophia. Puis les choses se sont accélérées avec notamment la décision par Raymond Barre le 8 février 1980 de la décentralisation d'une partie substantielle de l'Inria de la région parisienne vers Sophia Antipolis avec comme premier directeur du site Inria² Pierre Bernhard, qui était jusqu'alors directeur du Centre d'Automatique et d'Informatique (CAI) de l'École des mines de Paris, créé à l'initiative de Pierre Laffitte. D'autres grands noms du numérique arrivèrent ensuite, comme Digital Equipment ou France Telecom mais aussi des entreprises du hardware (Philips, Infineon Intel, ARM, Cadence, etc.) sur la spécialisation de conception de *chips* initialement pour les réseaux sachant que Texas Instruments et IBM étaient déjà implantés à quelques encablures de Sophia. Avec l'arrivée de l'ETSI et du W3C (hébergé par Inria), la technopole prit une couleur réseau au point de devenir la Telecom Valley grâce à la qualité de son écosystème autour du numérique qui continue d'innover. Et en 1989, l'université de Nice changea de nom pour devenir l'université Nice Sophia Antipolis³. Avec notamment la création du laboratoire d'informatique, signaux et systèmes de Sophia (I3S), l'université avait franchi le Var et le rêve de Pierre Laffitte faisait rêver le monde car bientôt d'autres technopoles voyaient le jour en reprenant les ingrédients du projet de Pierre Laffitte. Bien sûr, d'autres acteurs notamment autour des sciences de la vie et de la santé, des sciences de la Terre complètent le paysage de la technopole qui représentait en 2015, 38 000 personnes et 2 500 entreprises et établissements de recherche

2. Les locaux Inria furent livrés en 1983 et inaugurés début 1984 par Laurent Fabius.

3. Qui est devenue au 1^{er} janvier 2020, Université Côte d'Azur.

et d'enseignement supérieur répartis sur plus de 2 400 hectares et qui représentent environ 5,6 milliards d'euros de chiffre d'affaire du secteur privé.

Mais la contribution de Pierre Laffitte au développement de l'informatique et du numérique ne s'arrête pas là.

Sur la plan politique, Pierre Laffitte a été sénateur des Alpes Maritimes pendant plus de 23 ans (de 1985 à 2008). Infatigable défenseur de la recherche et de l'innovation, il a officié d'une part comme vice-président de l'office parlementaire d'évaluation des choix scientifiques et technologiques. On lui doit notamment en tant que co-rapporteur la proposition de loi en 1999 tendant à généraliser dans l'administration l'usage d'Internet et de logiciels libres⁴ qui est malheureusement toujours d'actualité. D'autre part, en tant que membre de la commission des affaires culturelles, dont il a été vice-président jusqu'en 2004, il a été longtemps chargé du rapport sur le budget de la recherche et l'on peut trouver sur le web de nombreuses traces de ses interventions pour demander plus de moyens pour la recherche notamment pour la recherche dans le numérique.

Car Pierre Laffitte a compris très tôt le caractère transformant du numérique par sa nature transversale qui touche tous les secteurs du savoir, de l'économie et de la culture et de l'éducation. Le numérique incarnait le mieux sa vision de fertilisation croisée car le numérique plus que d'autres secteurs est affaire d'écosystèmes.

Par exemple, Pierre Laffitte s'est fortement mobilisé lors de la création des pôles de compétitivité et la création du pôle « Solutions communicantes sécurisées (SCS) » lui doit beaucoup et il en fut même le premier président. Je me souviens également de son enthousiasme lors de la préparation d'une candidature d'un projet de l'Institut de recherche technologique (IRT) des premiers appels des investissements d'avenir, réponse qui avait été bâtie sur une vision d'un numérique transverse fondée sur la modélisation et les capacités de calcul et d'apprentissage (l'IA n'était pas encore à la mode) qui permettait de rassembler des entreprises diverses comme Thales Underwaters Systems, Amadeus, CS, Thales Alenia Space, Orange, Veolia, DCNS, Aéroports de la Cote d'Azur, etc. et des chercheurs des universités de la région Sud/Paca, du CNRS, de MINES ParisTech et d'Inria. Il avait vaillamment défendu le projet devant un jury qui *in fine* ne raisonnait que par filière industrielle et ne comprenait pas comment un assemblage d'industriels spécialisées en acoustique sous-marine, en spatial, dans le transport aérien, etc. pouvait fonctionner... C'était en 2011, Pierre Laffitte, malgré ses 86 ans, avait une vision très claire de ce qui pouvait les unir.

Jusqu'au bout Pierre Laffitte a porté des projets et pensé le futur. Bien sûr, le développement durable et le réchauffement climatique le préoccupaient ainsi que les enjeux géopolitiques de blocs et c'est à ce titre qu'il s'était impliqué dans les initiatives autour de la Méditerranée⁵. Lors d'une des dernières réunions que nous avons

4. <https://www.senat.fr/leg/pp199-117.html>.

5. <http://euro-mediterranee.blogspot.com/2009/02/pierre-laffitte-charge-de-la-creation.html>.

partagée en (2015-2016), il poussait pour mobiliser les acteurs autour des questions de cybersécurité en essayant de monter un axe franco-allemand. L'annonce du plan d'accélération cybersécurité, il y a quelques semaines, témoigne de la justesse de sa vision alors qu'il avait plus de 90 ans.

Bien sûr, Pierre Laffitte n'avait pas toujours raison et de son esprit créatif sortaient beaucoup d'idées et de projets et même s'il convenait de faire un peu de tri, la réflexion qu'ils permettaient d'enclencher ou le dynamisme intellectuel qu'ils impulsaient étaient toujours une source d'échange et d'enrichissement.

Pierre Laffitte nous a quitté le 7 juillet 2021. Il convient aux héritiers qui en ont fait l'éloge et qui veulent poursuivre l'œuvre de compléter Sophia Antipolis, la cité de la sagesse la bien nommée plus d'un demi-siècle après sa création, par un développement culturel plus important. En effet, pour Pierre Laffitte la fertilisation croisée tournée vers l'innovation prenait aussi ses racines dans un projet culturel présent dès le départ dans les fondements de la vision (un amphithéâtre de type romain avait été construit dès le début). Il n'y avait pas pour le père fondateur à mettre de barrière ni d'échelle de valeur entre culture et technique, schisme bien français déjà dénoncé par Gilbert Simondon⁶. Le numérique est peut-être une chance pour y arriver.

6. <https://www.lemonde.fr/blog/binaire/2020/04/05/informatique-culture-et-technique-le-schisme-de-simondon/>.



Turing à la plage — L'intelligence artificielle dans un transat

lu par Nathalie Revol¹

Turing à la plage — L'intelligence artificielle dans un transat

Rachid Guerraoui, Lê Nguyễn Hoang

collection À la plage, Dunod

juin 2020, 208 pages.

Le titre de ce livre pourrait laisser croire à une nouvelle biographie d'Alan Turing dans la lignée du film « *Imitation game*² » ou de la pièce « La machine de Turing³ » or il n'en est rien. Le sous-titre, « L'intelligence artificielle dans un transat », a toute son importance : il s'agit d'une introduction, très accessible, à l'intelligence artificielle. Le lien avec



1. Université de Lyon, Inria; texte publié par Interstices, <https://interstices.info/autour-dalan-turing/>.

2. <https://interstices.info/turing-les-images-dune-imitation/>.

3. <https://interstices.info/autour-dalan-turing/#la-machine-de-turing-theatre-BSoles>.

Turing ? Des références permanentes à ses écrits, des citations

qui mettent en avant l'aspect précurseur de Turing, des extraits de ses articles qui parlent de l'informatique telle que nous la connaissons, ses questionnements qui sont toujours d'actualité.

Le livre commence avec les notions d'algorithme⁴ et de machine et des dispositifs tout d'abord mécaniques puis électroniques qui réalisent ces algorithmes. Très vite, il aborde la formalisation de la notion d'algorithme et ses limites — qui ont constitué le cœur des travaux d'Alan Turing avant guerre, avec en particulier la fameuse machine de Turing⁵, modèle théorique d'algorithme. Les progrès pratiques, en termes de puissance de calcul, y sont présentés.

De plus en plus souvent, les performances des intelligences artificielles nous surprennent, comme nous le montre le livre, que ce soit en battant des champions aux jeux d'échecs ou de go, ou en reconnaissant des chats dans des images. Le livre présente aussi, rapidement, la notion de complexité⁶ des problèmes et les questions encore ouvertes en ce domaine, comme « P versus NP » ou « P versus NC »⁷.

Turing avait anticipé le fait que, quand la difficulté des tâches à résoudre augmente, la difficulté à écrire les algorithmes correspondants augmente au point que les algorithmes deviennent intraitables par un humain et il avait déjà proposé, en 1950, le principe des *learning machines*, principe sous-jacent aux algorithmes auto-apprenants, très présents dans l'intelligence artificielle que nous connaissons aujourd'hui.

Se pose alors la question de savoir distinguer une intelligence humaine d'une intelligence artificielle : du célèbre « test de Turing⁸ » publié en 1950 à d'autres expériences, de l'art avec ce que cela suppose de créativité aux algorithmes inspirés par la nature comme les automates cellulaires ou les algorithmes génétiques, différentes approches sont exposées.

Avant de conclure sur la « théorie hérétique » de Turing que nous vous laissons le plaisir de (re-)découvrir, les auteurs présentent aussi les facettes inquiétantes de l'utilisation des intelligences artificielles, qu'il s'agisse d'usages malveillants ou d'effets secondaires non anticipés et non désirés.

Ce livre peut tout à fait se lire à la plage, dans un transat, dans un hamac ou sous un plaid au coin du feu : il aborde très clairement les notions fondamentales de l'intelligence artificielle, avec une grande diversité et une grande pertinence dans les exemples choisis. S'il faut toutefois mettre un léger bémol à cet enthousiasme, on pourra parfois regretter une transposition de certains résultats de l'informatique à

4. <https://interstices.info/quest-ce-quun-algorithme/>.

5. <https://interstices.info/comment-fonctionne-une-machine-de-turing/>.

6. <https://interstices.info/glossaire/complexite/>.

7. <https://interstices.info/la-theorie-de-la-complexite-algorithmique/>.

8. https://fr.wikipedia.org/wiki/Test_de_Turing.

la vie de tous les jours, comme par exemple avec le choix de cette citation de Scott Aronson au sujet de la question « P versus NP » : « *Si $P = NP$, alors le monde est un endroit profondément différent de ce qu'on imagine habituellement. Il n'y aurait aucune valeur spécifique au saut créatif, aucun fossé séparant le fait de résoudre un problème et celui de reconnaître la validité d'une solution trouvée. Tous ceux capables d'apprécier une symphonie seraient Mozart; tous ceux capables de suivre un raisonnement étape par étape seraient Gauss.* » Certes les auteurs ne sont pas responsables de cette citation, mais ils ont choisi de relayer cette extrapolation, peut-être un peu excessive.

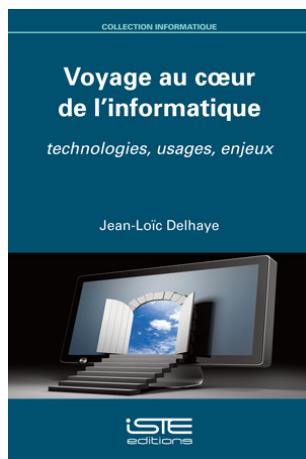


Voyage au coeur de l'informatique : technologies, usages, enjeux

lu par Michel Raynal¹

Voyage au coeur de l'informatique : technologies, usages, enjeux
Jean-Loïc Delhaye, ISTE editions
avril 2021, 238 pages.

Dans son ouvrage « Voyage au coeur de l'informatique », Jean-Loïc Delhaye nous offre un panorama de l'informatique pour « l'honnête homme/femme » du vingt-et-unième siècle. Dans ce domaine visant le « grand public » (trop souvent noyé sous les ouvrages essentiellement technologiques), l'auteur réussit le tour de force (en 230 pages) d'être toujours compris par le lecteur, tout en lui apprenant des choses qu'il ne connaissait peut-être pas ou qu'il connaissait mal. Pour cela, Jean-Loïc Delhaye articule son exposé sur ce qu'il appelle les trois piliers de l'informatique à savoir : le matériel, les moyens de communications (c'est-à-dire les réseaux) et le logiciel. Dans chaque cas, un historique du domaine est présenté ; ce qui permet au lecteur de mieux saisir les évolutions passées qui conditionnent notre présent.



1. Université de Rennes IRISA, Inria, CNRS, France, Department of Computing, Polytechnic University, Hong Kong.

Des titres de chapitres ou de sections incitent aussi le lecteur à une vision nuancée de certains domaines (par exemple « de l'élément binaire à l'intelligence », ou « intelligence ou statistiques »). Coté « usages », l'ouvrage présente des domaines applicatifs qui sont et vont devenir de plus en plus importants dans notre vie quotidienne, à savoir les robots, la réalité virtuelle/augmentée, la santé et la ville futée. L'auteur n'élude pas les questions morales et sociétales que soulèvent les applications de l'informatique ; un chapitre entier leur est consacré. Celui-ci aborde les dangers possibles pour la démocratie et la vie privée, la fracture numérique, la puissance actuelle des grandes firmes, etc. En clair, un livre agréable à lire qui tout à la fois nous apprend des choses et nous incite à la réflexion.



Une nouvelle approche au traçage numérique des contacts (application COVI au Québec)

Tegan Maharaj¹ et Yoshua Bengio²

Le traçage manuel des contacts est l'une des principales méthodes de gestion d'une épidémie. Lorsqu'un cas positif d'une maladie est identifié, ce processus nécessite qu'un responsable de la santé publique demande au patient de se rappeler tous les contacts qu'il a eus avec d'autres personnes. Le responsable tente ensuite de joindre chaque personne à risque pour l'avertir de s'isoler afin de réduire le risque de propagation de la maladie. Ce type de traçage manuel des contacts a été un élément crucial de la stratégie de santé publique de nombreux pays dans leurs efforts pour endiguer la COVID-19. Toutefois, l'ampleur de la pandémie dépasse parfois les ressources des responsables de traçage manuel. C'est une des raisons pour lesquelles le traçage numérique des contacts (TNC), qui envoie un message à un autre téléphone par l'entremise d'un protocole de messagerie protégée, a été étudié comme outil complémentaire à la méthode manuelle standard de traçage pouvant s'adapter à de vastes populations et à un grand nombre de cas. La méthode standard de traçage numérique repose sur des résultats diagnostics binaires (positifs ou négatifs) envoyés sous forme de messages sécurisés et anonymes entre téléphones intelligents pour ensuite fournir des recommandations binaires (mise en quarantaine ou pas).

COVI est une application mobile qui a été créée par une équipe de recherche interdisciplinaire dirigée par Yoshua Bengio de l'université de Montréal. COVI (cf. fi-

1. Docteur en informatique, école Polytechnique de Montréal.

2. Professeur en informatique à l'université de Montréal et récipiendaire du prix ACM Turing en 2019, yoshua.bengio@mila.quebec.

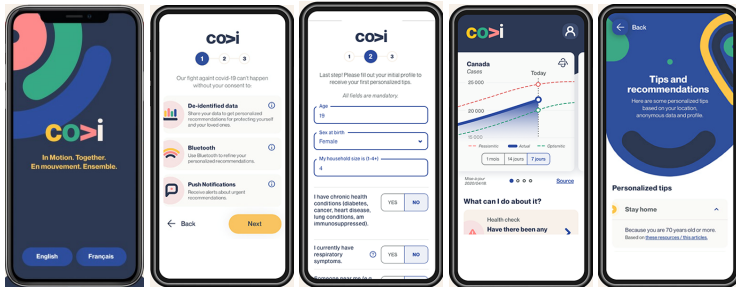


FIGURE 1. Application COVI au Québec.

gure 1) est une application de traçage numérique basée sur une approche novatrice qualifiée de traçage *proactif* des contacts. Ce nouveau type de traçage diffère du traçage binaire des contacts de deux façons importantes. Premièrement, il fait appel à de nombreuses sources d'information autres que les résultats positifs ou négatifs des tests (y compris les problèmes de santé préexistants, les symptômes déclarés par l'utilisateur et les habitudes de contacts répétés) afin de prédire le risque individuel d'infection par la COVID-19. Deuxièmement, grâce à ces sources d'information plus riches, il est en mesure de faire des recommandations graduées (non binaires) et personnalisées.

En formulant ces recommandations plus nuancées et plus ciblées de manière graduée et plus tôt, le traçage proactif des contacts devrait permettre d'atteindre le même endiguement de la propagation de la maladie avec moins de restrictions sur les activités habituelles des citoyens. Cela représente un potentiel important en matière de santé publique, dans la mesure où la propagation de la maladie pourrait être minimisée en sauvant davantage de vies. Ainsi, il serait moins nécessaire de recourir à un confinement total, qui a des retombées importantes sur l'économie et sur le bien-être et la santé mentale des citoyens.

Les applications de traçage numérique sont plus efficaces lorsque le taux d'adoption (la proportion de la population qui utilise l'application) est plus élevé. COVI n'est actuellement pas disponible en tant qu'application autonome, car les *App Stores* limitent les applications de traçage des contacts à celles sanctionnées par les gouvernements nationaux afin d'encourager une plus grande adoption. Cependant, le développement de COVI est désormais finalisé et l'application est prête à l'utilisation. Les codes et modèles pour le traçage proactif des contacts ont été examinés par des pairs de la communauté scientifique et sont disponibles en code source ouvert³, de sorte qu'ils pourraient être intégrés à toute application de traçage existante.

3. <https://github.com/mila-iqua/COVI-Mobile>.

Nous décrivons d'abord l'application du point de vue de l'utilisateur, puis son fonctionnement, notamment la manière dont les messages échangés entre les téléphones restent sécurisés et anonymes, et les raisons pour lesquelles l'utilisation de l'apprentissage automatique fait en sorte que les informations personnelles sensibles de l'utilisateur n'aient jamais besoin de quitter son téléphone. Nous décrivons ensuite le simulateur que l'équipe COVI a construit pour développer et évaluer les algorithmes de traçage des contacts proactifs, et détaillons deux algorithmes proactifs différents : une heuristique basée sur des règles et un prédicteur basé sur l'apprentissage automatique. Nous résumons les expériences que l'équipe COVI a menées en comparant ces méthodes et d'autres méthodes de traçage numérique des contacts, qui montrent le potentiel du traçage proactif des contacts pour réduire la propagation de la maladie avec des mesures moins restrictives. Nous concluons par une réflexion sur l'avenir du traçage numérique des contacts et sur le rôle que les méthodes proactives pourraient jouer dans la gestion des épidémies.

Utilisation de l'application COVI

Existe-t-il un visuel de l'application ? Lorsque vous ouvrez l'application pour la première fois, il vous sera demandé de fournir certaines informations sur vos antécédents médicaux (par exemple, votre âge, vos conditions médicales préexistantes) et vos symptômes actuels, le cas échéant. La saisie de ces informations est volontaire et n'est pas obligatoire pour utiliser l'application, mais leur fourniture améliorera l'exactitude des recommandations personnalisées. En raison du mode de fonctionnement du traçage proactif des contacts (tel que décrit ci-après dans la section expliquant l'entraînement du réseau neuronal), aucune de ces données n'a à quitter l'appareil de l'utilisateur. La seule information échangée entre les téléphones est un niveau de risque anonymisé et non traçable au téléphone de l'utilisateur. Les utilisateurs participant ont le choix de consentir ou pas à envoyer une version anonymisée et cryptée de leurs données qui seront conservées sur un serveur sécurisé doté de protections adéquates, où elles seront utilisées pour améliorer la précision du traçage proactif des contacts. Cette démarche est entièrement volontaire et n'est pas obligatoire pour utiliser l'application.

Après avoir configuré ces préférences, l'application fournit des informations générales sur la COVID-19 et des ressources pour votre région, ainsi que des recommandations personnalisées. Vous pouvez consulter l'application aussi souvent que vous le souhaitez, mais il n'est pas nécessaire de la garder ouverte pour qu'elle fonctionne. Vous recevrez une notification si les recommandations changent, que ce soit en raison de nouvelles politiques gouvernementales, de recommandations générales pour votre région ou de prédictions faites à partir de la recherche de contacts. À tout moment, si vous ressentez des symptômes ou obtenez des résultats de tests, vous devez les entrer dans l'application pour obtenir des recommandations actualisées et pour permettre une estimation précise des risques.

La messagerie anonyme entre les téléphones

Lorsque deux téléphones munis de l'application se trouvent à moins de deux mètres l'un de l'autre pendant 15 minutes, un contact est enregistré. Dans ce cas, les téléphones échangent des clés sécurisées via Bluetooth que les téléphones peuvent utiliser plus tard pour envoyer des mises à jour sécurisées sur le niveau de risque de ce contact. Ce niveau de risque se traduit par une recommandation personnalisée provenant du téléphone, qui aide à prévenir la propagation de la maladie (par exemple, éviter de faire les courses aujourd'hui). Le format exact du message et la sécurité de la communication dépendent du protocole de communication. Bien que COVI puisse être utilisée avec n'importe quel protocole de communication, incluant celui développé dans le cadre d'un partenariat entre Google et Apple intitulé *Google/Apple Exposure Notification* (GAEN). Il est important de noter que pour garantir la protection de la vie privée et la sécurité des utilisateurs, ces messages sont quantifiés selon une précision de 4 bits, ce qui donne 16 niveaux de risque possibles. L'échange de clé chiffrée est valide pour une période de 14 jours. Tout changement de niveau de risque pendant cette période déclenche un message de mise à jour du risque sécurisé pour toutes les clés valides, couvrant les 14 derniers jours. Par souci de confidentialité avant qu'un message quitte le téléphone d'un utilisateur, l'application émet des lots de données chiffrées à intervalles périodiques, soit quatre fois par jour.

Le calcul des niveaux de risque

Avec le traçage proactif des contacts, le risque correspond à l'infectiosité prédite (la probabilité qu'une personne infecte d'autres personnes). Ce calcul est prédit localement sur votre téléphone par un réseau de neurones pré-entraîné. Afin de faire des prévisions d'infectiosité, ce réseau neuronal recueille des informations sur les conditions médicales préexistantes, les symptômes déclarés par l'utilisateur, les habitudes de contacts répétés et les messages de mise à jour des risques des 14 derniers jours. En mettant à jour les prévisions pour les jours passés à partir de nouvelles informations, les recommandations peuvent être proactives (donc basées sur une prédiction du risque), plutôt que réactives (tels que le sont les traçages manuels et binaires). Toujours à partir du téléphone de l'utilisateur, l'infectiosité prédite est convertie en un niveau de risque. Si ce niveau de risque diffère de celui calculé à la fin du dernier intervalle, un message de mise à jour avec le nouveau niveau de risque est transmis à toutes les clés chiffrées avec lesquelles l'utilisateur a été en contact.

Comment le réseau neuronal est-il entraîné ?

Le réseau de prévision de l'infectiosité est formé hors-ligne sur des données simulées par un modèle épidémiologique. Afin de générer des données d'entraînement de haute fidélité, nous avons collaboré avec des épidémiologistes de premier plan pour mettre au point un simulateur détaillé que nous appelons COVI-AgentSim⁴,

4. <https://github.com/mila-iqia/COVI-AgentSim>

décrit ci-dessous. L'objectif est de générer des données qui ressemblent à celles que nous obtiendrons à partir d'une application de gestion des risques de la COVID-19, qui suit les symptômes et les contacts potentiellement risqués des personnes avec une conception privilégiant le respect de la vie privée. Avec les échantillons générés par le simulateur, nous disposons de l'information complète sur l'état de l'infection et les événements de contagion qui sont importants pour comprendre et prédire la propagation de la maladie, et ce, pour chaque agent (humain simulé) dans la simulation. Grâce à ces données, nous pouvons entraîner les modèles d'apprentissage automatique à prédire des variables non observées comme l'état d'infection, les événements de contagion, etc., à partir de variables observées comme les symptômes, les conditions préexistantes et les rencontres avec d'autres personnes utilisant l'application. En formant un prédicteur initial sur des données simulées qui incorporent des connaissances épidémiologiques et les contraintes probabilistes associées, nous sommes en mesure d'avoir des prédicteurs de risque plus précis que ce qui aurait pu être obtenu par un apprentissage sur des données réelles uniquement, tout en respectant la vie privée des personnes.

Le simulateur COVI-AgentSim

Le simulateur de notre application s'inspire d'un modèle basé sur les agents, classique en épidémiologie et écologie. Il diffère des modèles basés sur les populations qui utilisent des équations différentielles uniquement pour modéliser les statistiques de l'infection. Le modèle basé sur les agents simule les caractéristiques et les comportements des individus ; chaque agent représente les aspects ciblés et virtuels d'une personne susceptible d'être infectée et contagieuse, ce qui motive ce choix pour notre projet. Les deux types de modèles peuvent être décrits comme étant des modèles à compartiments s'ils captent l'évolution de la proportion de la population dans un nombre fini d'états possibles⁵. COVI-AgentSim considère les états Susceptible (S), Exposé (E), Infecté et Récupéré (R) du modèle comportemental courant. Le simulateur est développé en Python et basé sur le package Simpy⁶. Une population humaine dont les caractéristiques individuelles suivent des statistiques régionales est créée dans une ville simulée. Chaque individu se déplace selon des trajectoires générées que nous réglons pour qu'elles correspondent aux données démographiques de la région visée. Une partie de ces individus ont la maladie et, lorsqu'ils se déplacent dans la ville (passant du temps dans des lieux comme leur domicile, leur travail, les transports en commun, des commerces, des hôpitaux, des centres de soins de longue durée, etc.), ils peuvent s'infecter les uns les autres, présenter des symptômes, être hospitalisés, et ainsi de suite. Nous suivons la propagation de la maladie au moyen de plusieurs paramètres (le taux de reproduction R, le taux d'atteinte, etc.), que nous réglons pour qu'ils correspondent aux données réelles et aux résultats d'un modèle

5. <https://docs.idmod.org/projects/emod-hiv/en/latest/model-seir.html>.

6. <https://simpy.readthedocs.io/en/latest>.

à compartiments (SEIR) ajusté selon les données provenant de la ville de Wuhan et adapté aux caractéristiques sociodémographiques de la population canadienne. Le simulateur produit des séquences de rencontres entre les agents, accompagnées de renseignements sur la transmission de la maladie. Nous nous servons de ces séquences pour créer un ensemble de données destiné aux modèles d'apprentissage automatique afin de prédire le risque individuel à partir de variables observées, comme les symptômes, les problèmes de santé préexistants et les lieux visités par la personne.

Quelques informations sur le simulateur :

- la ville simulée est dotée d'une structure de graphe pour les lieux — notamment les domiciles, les magasins, les parcs, les hôpitaux (dont les unités de soins intensifs), les centres d'hébergement et les différents modes de transport — tels que la marche, le vélo, le métro et l'autobus, le covoiturage ou le taxi et la voiture. À chaque lieu et à chaque moyen de transport sont associées des propriétés relatives à la capacité et à la transmission de la maladie ;
- toutes les données démographiques et les statistiques paramétrées par région peuvent être définies en fournissant les fichiers de configuration appropriés ; actuellement, le simulateur est configuré pour Montréal au Canada ;
- chaque individu possède des caractéristiques individuelles : âge, sexe, problèmes de santé préexistants, degré de prudence, utilisation ou non de l'application, fréquence du port d'un masque, lieu de travail, etc. Ces caractéristiques sont échantillonnées aléatoirement selon les données démographiques pour le Canada. Les individus sont dotés de caractéristiques épidémiologiques, telles que la charge virale, la contagiosité et la progression des symptômes durant la maladie, qui dépendent de ces caractéristiques individuelles (voir ci-dessous pour plus de précisions) ;
- de nombreux événements se produisent tandis que les individus se déplacent et que le temps passe :
 - une rencontre se produit lorsque deux individus sont suffisamment rapprochés dans l'espace et le temps. Dans le cas du protocole GAEN, cela signifie une distance de 2 mètres pendant 15 minutes. Si l'un des individus est contagieux, il infectera l'autre selon une probabilité proportionnelle à sa contagiosité,
 - si les individus consignent leurs symptômes dans l'application ou obtiennent un résultat positif ou négatif à un test, un nouveau niveau de risque est calculé. Les centres de dépistages présentent une capacité fixe invariable, et les différents types de test ont des taux de faux négatifs différents en fonction du temps, établis à partir de la littérature. Les individus enregistrent leurs symptômes avec des probabilités variables en fonction des caractéristiques individuelles,
 - Les individus peuvent attraper un rhume, une grippe ou des allergies saisonnières, pondérés selon les statistiques régionales enregistrées sur ces

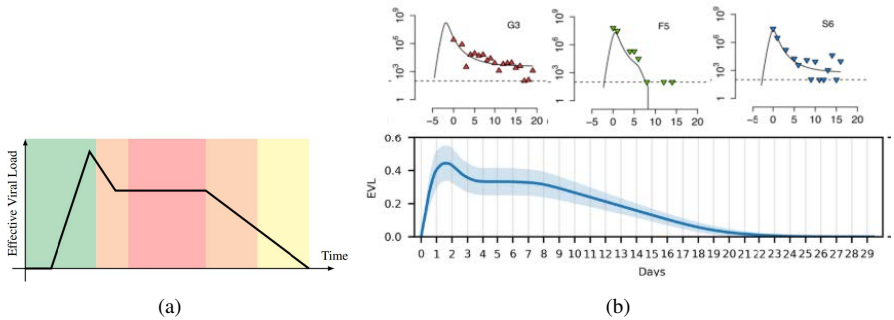


FIGURE 2. Un diagramme schématique des symptômes (a) où les couleurs dénotent la gravité et une courbe réelle (b) moyennée (en bas) avec quelques cas individuels (en haut).

maladies. Il en résulte une répartition plus réaliste des symptômes (car la COVID-19 n'est pas la seule cause de symptômes). De plus, nous ajoutons du bruit (symptômes aléatoires), afin de résister à l'entrée accidentelle de symptômes.

Les détails épidémiologiques

L'horodatage de l'infection désigne le moment précis où un individu est infecté. Les individus peuvent être infectés en fonction d'un lieu (par exemple, si un individu très contagieux s'y trouvait peu de temps auparavant, ou, ce qui est plus probable, lors d'une rencontre avec une autre personne). Nous suivons la source d'exposition pour chaque individu infecté.

La charge virale est la quantité de virus mesurée chez un humain infecté. Sur la base d'études sérologiques qui mesurent cette quantité, nous modélisons une quantité que nous appelons la charge virale effective. Celle-ci saisit l'interaction du virus avec le système immunitaire de l'hôte dans un nombre compris entre 0 et 1 qui représente la sévérité de l'infection actuelle chez la personne. Cette quantité est modélisée en tant que fonction linéaire par morceaux comprenant cinq stades de l'évolution de la maladie : incubation (charge virale effective 0), augmentation, diminution, plateau et rétablissement, illustrés dans la figure 2. Dans cette figure, tous les chiffres sont relatifs aux jours d'incubation (apparition des symptômes), soit en moyenne 5,8 jours après l'exposition :

- l'augmentation commence dès le début de la contagiosité, en moyenne 2,3 jours avant les jours d'incubation, et atteint un pic de 0,7 jour avant les jours d'incubation ;

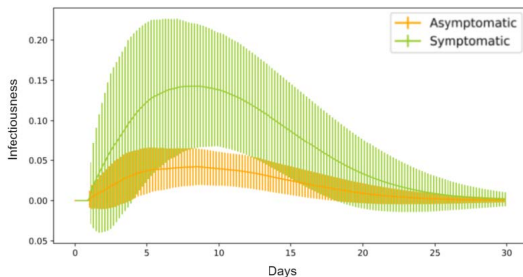


FIGURE 3. Infectiosité moyenne des individus.

- le plateau dure un certain nombre de jours et est échantillonné selon une loi normale ayant pour espérance une durée de 5 jours ;
- la diminution, ou le rétablissement, dure un certain nombre de jours d'incubation et est échantillonné à partir d'une loi normale ayant pour espérance une durée de 14 jours.

La contagiosité est proportionnelle à la charge virale, mais dépend de caractéristiques telles que le fait d'être asymptomatique ou immunodéprimé, ou de porter un masque, de tousser, et ainsi de suite. L'infectiosité moyenne, ventilée par individus symptomatiques et asymptomatiques, est illustrée dans la figure 3. La progression des symptômes est tributaire de la charge virale. Pour chacun des cinq stades de la figure 2(a), les symptômes sont échantillonnés en fonction de leur prévalence moyenne chez les patients atteints de la COVID-19. Les symptômes débutent en moyenne 5,8 jours après l'exposition (jours d'incubation). L'application que nous avons développée modélise environ une vingtaine de symptômes.

Il y a dans le modèle une probabilité d'environ 29 % qu'une personne infectée soit asymptomatique (ce qui réduit sa contagiosité à 10 % de ce qu'elle serait normalement), une probabilité d'environ 15 % (modulée par âge) qu'elle soit très malade (état nécessitant une hospitalisation) et une probabilité d'environ 30 % qu'elle soit extrêmement malade (état nécessitant un séjour aux soins intensifs). Il y a une probabilité de 0,2 % que la personne ne se rétablisse jamais (également modulé par âge). Le simulateur pourrait modéliser la réinfection, mais ne le fait pas pour le moment à cause de l'incertitude qui plane toujours sur la fréquence de ce phénomène. Nous validons plusieurs paramètres à partir de simulations afin de calibrer le simulateur aux données réelles. Les paramètres les plus importants sont résumés dans la figure 4. Les chiffres indiquent une moyenne μ en jours et un écart-type σ , calculé sur 10 tirages aléatoires sur une population de 3 000 habitants. Il faut souligner que ces statistiques sont le résultat de nombreux processus qui se déroulent au sein du modèle épidémiologique. Il n'existe pas de paramètres qui codent spécifiquement

Metric	Simulator $\mu \pm \sigma$	Reference
Incubation	5.505 \pm 0.01	5.807 [34]
Infectiousness onset	3.357 \pm 0.016	2.3 [8]
Recovery	18.886 \pm 0.035	14 [8]
Generation time	4.341 \pm 0.018	3.99 [36]
Daily contacts	16.355 \pm 0.137	13.4 [37]
Presymptomatic transmission	0.563 \pm 0.011	0.44 [8]
Asymptomatic transmission	0.226 \pm 0.004	0.29 [9]

FIGURE 4. Mesures obtenues par le simulateur et leurs valeurs réelles.

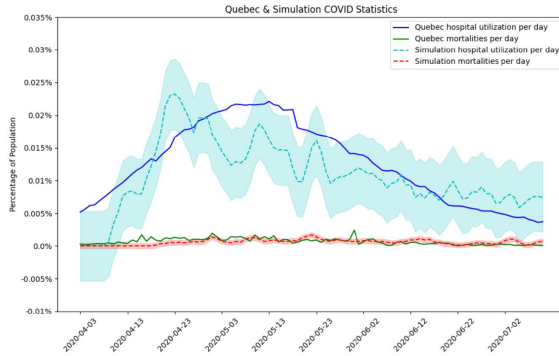


FIGURE 5. Évolutions réelle et simulée du nombre d’admissions en hôpital et du nombre de morts.

ces valeurs. Nous montrons également un ajustement qualitatif aux données réelles des admissions à l’hôpital et du nombre de mortalités par jour dans la figure 5.

Générer les données d’apprentissage à partir du simulateur

Nous utilisons un processus de génération aléatoire pour obtenir un ensemble de données qui représente un degré réaliste de variation entre les paramètres du simulateur. Pour chaque simulation réalisée, les paramètres sont choisis au hasard parmi une portée réaliste et prédéfinie. Cela permet d’utiliser le simulateur pour couvrir une grande variété de paramètres, sans avoir besoin de couvrir de manière exhaustive tous les scénarios possibles, ce qui serait prohibitif en termes de calcul. À partir de cet éventail de données d’apprentissage, les méthodes d’apprentissage automatique sont capables de généraliser (interpoler) très efficacement à travers le domaine d’apprentissage.

Le traçage proactif des contacts : aperçu du modèle prédictif

Le diagramme de la figure 6 donne un aperçu du traçage proactif des contacts (TPC). La partie gauche de la figure présente la transmission de messages anonymes

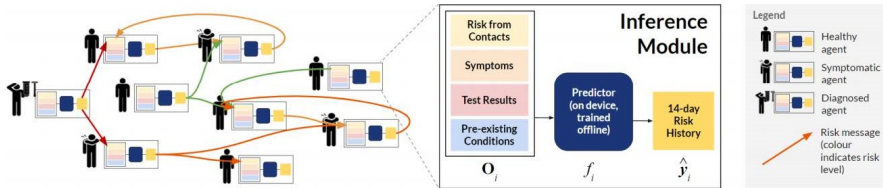


FIGURE 6. Portrait du traçage proactif des contacts.

à risque gradués (non-binaires) entre utilisateurs. L’application de chaque utilisateur contient un module d’inférence (encadré) qui, à partir d’observations O_i et un modèle prédictif f_i pré-entraîné estime le risque d’infection attendue de la personne \hat{y}_i pour chacun des 14 derniers jours. Des éléments anonymes sélectionnés à partir de ce vecteur de risque sont envoyés sous forme de messages aux contacts appropriés, ce qui leur permet de mettre à jour de manière proactive leur propre estimation de l’infectiosité attendue. Le module d’inférence doit prendre en compte un ensemble d’informations (il n’y a pas d’ordre entre les éléments de l’entrée), mais les prévisions ont lieu sur une fenêtre temporelle. Le TPC peut opérer avec n’importe quel réseau neuronal capable de faire face à ces contraintes. Nous expérimentons avec deux sortes de transformateurs à valeur fixe. Nous entraînons le modèle prédictif sur l’ensemble des données randomisées en utilisant une méthode de descente de gradient stochastique pour l’objectif supervisé de la prévision de l’infectiosité. L’entrée comprend des informations mises à jour dynamiquement : les messages de risques et des mises à jour de tous les contacts des 14 derniers jours, les symptômes déclarés que la personne éprouve actuellement, les résultats des tests qu’elle a obtenus, ainsi que des informations statiques sur ses conditions médicales préexistantes, son âge, etc. La cible est l’historique d’infectiosité avec la vérité terrain généré par le simulateur.

Un aspect important de cet entraînement est que les cibles (infectiosité) du modèle prédictif obtenues de la simulation dépendent des prévisions de l’application employée dans la simulation. Pour atténuer les effets de ce changement, nous effectuons plusieurs cycles de ce que nous appelons un réapprentissage itératif.

Comparaison des méthodes de traçage numérique des contacts

Nous comparons les modèles de réseaux neuronaux profonds à une simple régression linéaire sur les mêmes entrées. Cette comparaison révèle que les interactions non linéaires entre caractéristiques sont essentielles à la réalisation de bonnes prévisions.

Afin de déterminer si l’on pourrait obtenir une version plus simple de l’application qui ne nécessite pas d’ajustement aux données (et donc ne nécessiterait pas de

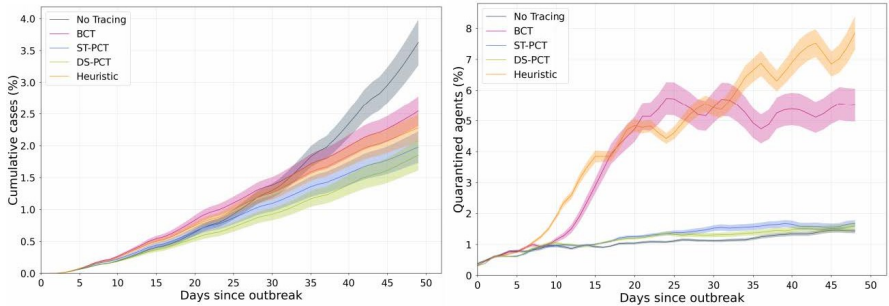


FIGURE 7. À gauche, le nombre cumulé de cas pour chaque méthode; à droite, les restrictions de déplacement pour les mêmes expériences (une fraction des quarantaines).

recueillir de données des téléphones pour améliorer l'application), nous avons développé une méthode heuristique fondée sur des règles. Le TPC avec apprentissage profond dépasse les méthodes fondées sur la régression linéaire et heuristique, mais l'heuristique fournit une base de référence étonnamment forte. Il est donc susceptible de se révéler utile pour un premier déploiement avant que suffisamment de données aient été recueillies ou pour éviter les coûts d'une infrastructure de ré-entraînement régulier des modèles.

La figure 7 montre les résultats quantitatifs du nombre de cas en fonction du temps en nombre de jours. L'absence de traçage (méthode la plus élevée au bout de 50 jours sur le graphique de gauche) est le scénario où aucun membre de la population ne dispose d'une méthode de traçage numérique des contacts, mais où la plupart des individus suivent les directives de mise en quarantaine lorsqu'ils obtiennent un résultat de test positif. Nous constatons que toutes les applications de traçage numérique peuvent faire mieux. Les méthodes proactives basées sur les réseaux neuronaux profonds (ST-PCT et DS-PCT) sont capables de réaliser cette diminution tout en imposant très peu de restrictions de déplacement (graphique de droite), voire à être proche du cas sans traçage (*No tracing*).

Le graphique de la figure 8 illustre le taux de reproduction R pour chaque méthode de traçage. On constate que toutes les méthodes de TNC sont capables de réduire le taux R en dessous de 1 (c'est-à-dire qu'elles permettraient à terme de freiner la propagation de la maladie), mais les méthodes ST-PCT et DS-PCT basées sur l'apprentissage profond sont les plus efficaces.

Le fait que les applications numériques de recherche des contacts ne soient efficaces qu'avec un grand nombre d'utilisateurs suscite de vives inquiétudes de la part des utilisateurs. Cependant, plusieurs études ont maintenant démontré le contraire,

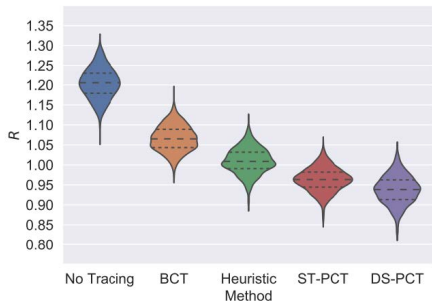


FIGURE 8. Indice de reproduction pour chaque méthode de traçage.

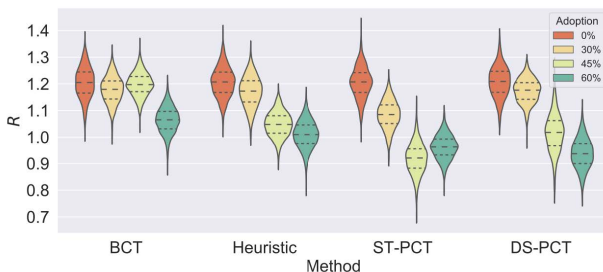


FIGURE 9. Analyse du taux d'adoption par les individus pour chaque méthode de traçage.

à savoir que les applications de traçage de contact numérique peuvent être utiles et sauver des vies même à des taux d'adoption faibles.

Nous avons effectué une analyse de la sensibilité des différentes méthodes au taux d'adoption, et nos résultats (cf. figure 9) confirment les conclusions des études précédentes : si des taux d'adoption plus élevés rendent ces applications plus efficaces, même des taux d'adoption faibles (à 30 %) représentent une amélioration comparée à l'absence de traçage.

L'avenir du traçage numérique des contacts

Le traçage numérique des contacts pourrait aider à prévenir des éclosions à l'avenir avec un déconfinement sécurisé. Cette recherche fournit également une base pour concevoir des applications de traçage proactif des contacts en vue de futures épidémies. Cette recherche est le fruit d'une collaboration entre des experts de nombreux

domaines. Nous espérons qu'elle pourra inspirer le type de progrès scientifique nécessaire pour résoudre certains des problèmes auxquels l'humanité est confrontée.



L'infection du damier

Jean-Paul Delahaye¹

La rubrique « Récréation informatique » propose une petite énigme algorithmique ou sur un thème de mathématiques discrètes susceptible d'intéresser un lecteur de 1024. La solution est donnée dans le numéro suivant.

Rappel et solution du problème précédent

LES PIÈCES MAGNÉTIQUES

On procède à des lancers successifs de pièces magnétiques dans une boîte à deux compartiments A et B. Au départ, il y a une pièce dans chaque compartiment. Quand on lance une nouvelle pièce magnétique, elle peut tomber dans le compartiment A ou dans le compartiment B. S'il y a x pièces dans A et y dans B, du fait de son magnétisme, la nouvelle pièce tombe dans A avec une probabilité $\frac{x}{x+y}$ et dans B avec une probabilité $\frac{y}{x+y}$. On lance ainsi 1000 pièces magnétiques en plus des deux pièces initiales.

Quelle est la probabilité pour que le nombre de pièces s'équilibre exactement entre A et B quand on a lancé les 1000 pièces ? Quelle est la probabilité pour que toutes les pièces lancées aillent en A ?

Indication : le résultat est simple.

1. Professeur émérite, université de Lille, campus scientifique, CRISAL UMR CNRS, 9189 Centre de recherche en informatique signal et automatique de Lille, bâtiment ESPRIT, 59655, Villeneuve d'Ascq Cedex France. E-mail : jean-paul.delahaye@univ-lille.fr.

SOLUTION.

Merci à Eric Wegrzynowski et Marilyn Rosselle qui m'ont fait parvenir la solution.

La solution est que toutes les répartitions possibles pour les 1000 nouvelles pièces lancées $(0, 1000), (1, 999), (2, 998), \dots, (1000, 0)$ ont la même probabilité d'être observées, soit $\frac{1}{1001}$ exactement. Le démontrer ne semble pas immédiat. Le raisonnement suivant, dû à Peter Winkler [1], est particulièrement rapide et astucieux. Il se base sur une modélisation de la boîte à l'aide d'un jeu de cartes.

On imagine un jeu de 1000 cartes blanches et une carte rouge. On va mélanger ces cartes d'une manière parfaite et, cela donnera la réponse à notre problème de pièces magnétiques ! On prend en main toutes les cartes, la rouge au-dessus, les 1000 blanches numérotées de 1 à 1000, en dessous. On pose la carte rouge sur la table. On prend la carte blanche numéro 1 et on la place au hasard en dessous ou au-dessus de la carte rouge (50 % de chances de la placer en dessous, 50 % de chances de la placer au-dessus). On prend la carte blanche numéro 2, on la place au hasard à l'une des trois places possibles dans le paquet de deux cartes posés sur la table (c'est-à-dire, on place la carte au-dessus des deux premières avec la probabilité $\frac{1}{3}$, entre les deux cartes déjà sur la table avec la probabilité $\frac{1}{3}$, et en dessous avec la probabilité $\frac{1}{3}$). On procède de même avec la carte blanche numéro 3 (quatre possibilités, chacune avec une probabilité $\frac{1}{4}$). Et ainsi de suite, jusqu'à la carte blanche numéro 1000. En procédant ainsi, on a parfaitement mélangé les 1001 cartes, car la méthode ne favorise aucun emplacement pour aucune carte. On peut vérifier par un petit calcul qu'une fois n cartes placées, chacune possède exactement une chance sur n d'être en position n .

Examinons maintenant ce qu'il se passe par rapport à la carte rouge quand on place la carte blanche numéro $n - 1$ dans le paquet de $n - 1$ cartes ($n - 2$ blanches et la rouge). Si on note $x - 1$ le nombre de cartes blanches sur la carte rouge et $y - 1$ le nombre de cartes blanches sous la carte rouge ($x + y = n$), en plaçant la carte numéro $n - 1$, on a une probabilité $\frac{x}{x+y}$ de la placer sur la carte rouge et $\frac{y}{x+y}$ de la placer sous la carte rouge. C'est exactement la même chose que lorsqu'on lance la pièce dans la boîte à deux compartiments A et B avec x pièces en A et y en B. Le procédé de placement des cartes blanches sous ou sur la carte rouge est parfaitement analogue à celui du remplissage des compartiments A et B de l'urne.

La répartition finale x pour A, y pour B, des 1000 pièces lancées dans l'urne (en ne prenant pas en compte les deux pièces initiales) est donc atteinte avec la même probabilité que la répartition de x cartes blanches au-dessus de la carte rouge et y cartes blanches en dessous de la carte rouge. Ces différentes probabilités sont aussi égales à celle que la carte rouge soit placée en position $x + 1$ quand on mélange parfaitement un paquet de 1001 cartes, c'est-à-dire $\frac{1}{1001}$. Toutes les répartitions sont

donc équivalentes : il y a exactement la même probabilité de placer toutes les nouvelles billes en A que d'en placer 500 en A et 500 en B, ou n'importe quelle autre répartition $(x, 1000 - x)$.

Ce problème est dû à George Polya [2]. Si vous avez des doutes sur le raisonnement, vous pouvez faire quelques expériences avec une urne (ou un jeu de cartes) en prenant 5 ou 10 pièces au lieu de 1000... ou en écrivant un programme.

Nouveau problème

L'INFECTION DU DAMIER

Un damier de taille carrée et de n cases de côté subit une infection dont la règle de fonctionnement est la suivante : si une case non infectée a au moins deux voisines infectées, elle l'est à la seconde suivante. Ne comptent comme voisines d'une case donnée que la case en dessous, la case au-dessus, la case à gauche, et la case à droite.

Si on suppose qu'à un instant donné toutes les cases d'une diagonale sont infectées alors on comprend que progressivement toutes les cases se trouveront infectées (cf. figure 1).

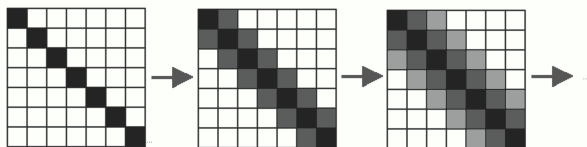


FIGURE 1. Infection progressive du damier.

Le mécanisme d'infection semble particulièrement violent et rapide, pourtant tout n'est pas possible, et même si cela semble paradoxal, pour infecter le damier de côté n dans son entier, il faut, qu'au départ, il y ait au moins n cellules infectées.

Pouvez-vous le démontrer ? Le plus étonnant dans ce problème est qu'un seul mot donne la démonstration recherchée.

Envoyez vos réponses à jean-paul.delahaye@univ-lille.fr. Les noms des premiers lecteurs à me donner la bonne réponse (et à la justifier) seront mentionnés dans le prochain numéro de 1024.

Références

- [1] Winkler Peter. Mathematical puzzles : a connoisseur's collection. CRC Press, 2003.
- [2] Polya, George. How to solve it : A new aspect of mathematical method. No. 246. Princeton university press, 2004.

ADHÉRER À LA SIF

La Société informatique de France est un espace de réflexion, de concertation sur les enjeux de l'informatique, mais aussi un espace d'actions, basé sur le travail de la communauté, qui vise à rassembler toutes celles et tous ceux pour qui faire progresser l'informatique est un métier ou une passion : enseignants, chercheurs, ingénieurs, industriels, consultants et étudiants.

Les adhésions sont valables 12 mois à compter de la date d'adhésion.

Personnes physiques

Tarif plein : 30 €

Tarif réduit : 15 €

- Membre d'un adhérent institutionnel de la SIF.
- CDD, CDI depuis moins de 2 ans, retraité.
- Membre d'une association partenaire, ou de l'ACM.

Gratuit : étudiants, doctorants et post-doctorants.

La SIF vous offre la possibilité d'effectuer le règlement de la cotisation directement en ligne.

Partenaires (Personnes morales)

Les associations partenaires, membres du Conseil des associations de la SIF, ne paient pas de cotisation. Les institutions telles que laboratoires, unités d'enseignement, ou entreprises, peuvent adhérer en tant que telles à la SIF. Il n'existe pas de tarif spécifique pour les adhérents institutionnels : en fonction de leur taille, de leur secteur d'activité, l'importance de l'effort ne se mesure pas de la même façon.

La SIF propose cinq niveaux de cotisation. Pour vous aider dans votre choix, vous trouverez ci-dessous une indication du tarif en fonction de la taille de l'adhérent institutionnel :

- Tarif 1 : 250 € (moins de 50 personnes)
- Tarif 2 : 500 € (de 50 à 100 personnes)
- Tarif 3 : 1000 € (de 100 à 150 personnes)
- Tarif 4 : 1500 € (de 150 à 200 personnes)
- Tarif 5 : 2000 € (au delà de 200 personnes)

Pour adhérer, vous devez contacter notre trésorier. Pour toute question, ne pas hésiter à contacter notre secrétariat.

Plus d'informations sur notre site internet :

<https://www.societe-informatique-de-france.fr>



B U L L E T I N

de la société informatique
de France



Institut Henri Poincaré,
11 rue Pierre et Marie Curie,
75231 Paris Cedex