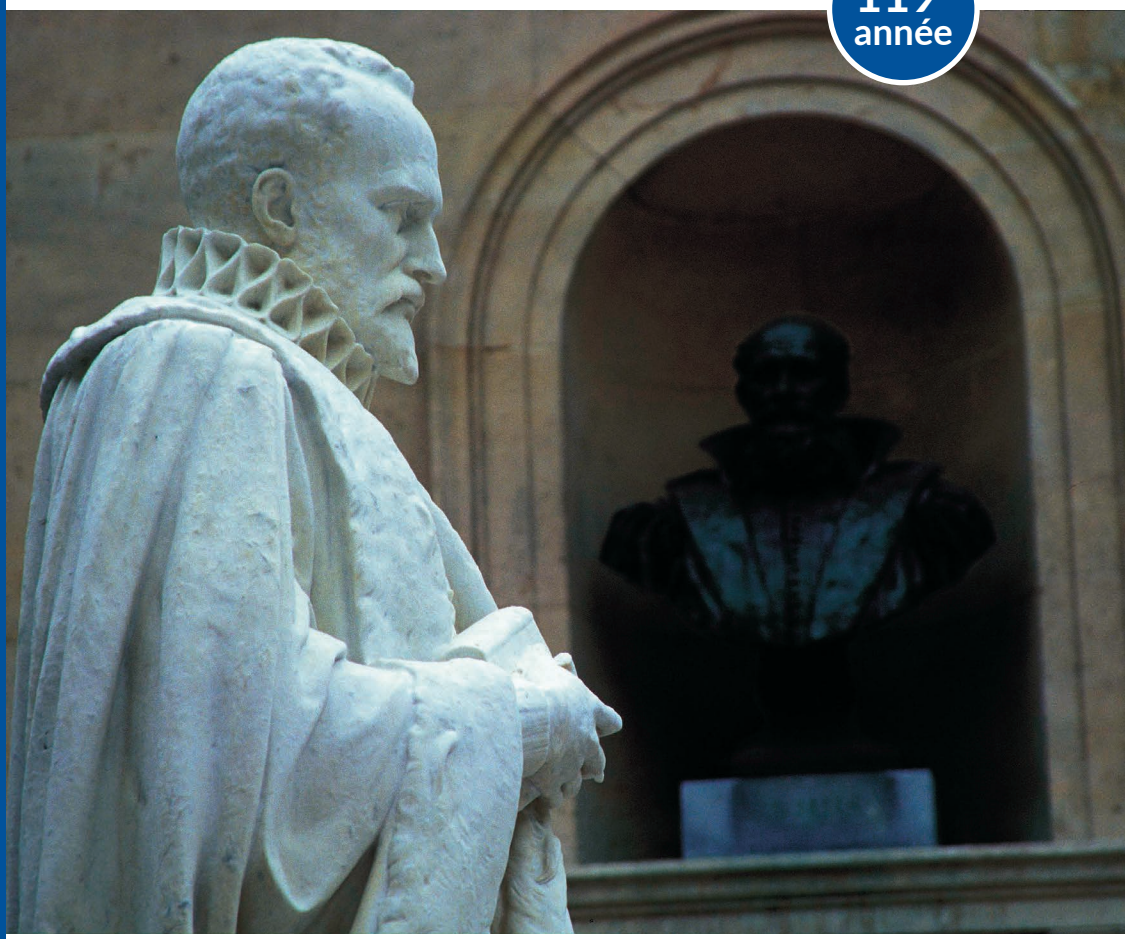
ANNUAIRE du **COLLÈGE DE FRANCE** 2018 - 2019

Résumé des cours et travaux

119e année



COLLÈGE DE FRANCE
1530

Algorithmes, machines et langages

Gérard Berry

Membre de l'Institut (Académie des sciences) et de l'Académie des technologies, professeur au Collège de France

Mots-clés : algorithmes, Esterel, langages de programmation, temps

La série de cours et séminaires « Où va l'informatique ? » est disponible, en audio et/ou en vidéo, sur le site internet du Collège de France (https://www.college-de-france.fr/site/gerard-berry/course-2018-2019.htm) ainsi que le colloque « L'imagerie à l'heure de l'IA : défis et opportunités » (https://www.college-de-france.fr/site/gerard-berry/symposium-2018-2019.htm) et la leçon de clôture « Plaidoyer pour les trajectoires non linéaires » (https://www.college-de-france.fr/site/gerard-berry/course-2019-02-26-16h00.htm).

Cours et séminaires – Où va l'informatique ?

Introduction

Cette année 2018-2019 a été la dernière de mon enseignement au Collège de France. Elle m'a permis de fermer la parenthèse ouverte en 2007-2008 sur la chaire annuelle Innovation technologique Liliane Bettencourt avec le cours « Pourquoi et comment le monde devient numérique ». Pour cette première présentation de l'informatique au Collège de France, j'avais choisi un cours grand public, avant de traiter des sujets plus techniques les années suivantes. Fermer la parenthèse consistait à revenir onze ans après à un cours destiné à un public large mais curieux et attentif, celui du Collège de France, pour faire un point sur l'état de l'informatique et tracer les lignes de force probables de son évolution.

Le paysage informatique a beaucoup changé depuis 2007 : le « numérique » a révolutionné le monde en s'appuyant sur la science informatique et sur l'ensemble

G. Berry, « Algorithmes, machines et langages », *Annuaire du Collège de France 2018-2019. Résumé des cours et travaux*, 119e année, Paris, Collège de France, 2022, p. 13-20, https://doi.org/10.4000/annuaire-cdf.16725.

des technologies qu'elle nourrit. Mais le grand public reste largement ignorant de la vraie nature de l'informatique et des causes de ses grandes évolutions. Elle est peu et mal décrites dans les médias, qui tendent à se concentrer tous ensemble sur un seul sujet à la fois, par exemple l'impression 3D, puis l'apprentissage automatique par réseaux de neurones profonds, une intelligence artificielle à laquelle on prête indifféremment des miracles ou des peurs qui n'ont que peu à voir avec la réalité, effectivement impressionnante. Bien d'autres sujets, tout aussi importants mais moins propices aux fantasmes, sont trop rarement discutés. De fait, la vulgarisation de l'informatique reste bien moins développée que celle de la physique ou de l'astronomie.

Or la société informatisée qui nous attend dépendra directement de nos choix conscients ou inconscients. La persistance d'une mauvaise compréhension des raisons de la puissance des mouvements actuels nous a longtemps conduits à subir servilement les choix faits par les autres. Pour faire des choix plus sensés, il faudra d'abord mieux comprendre. Mon objectif a donc été d'expliquer les ressorts de l'informatique moderne au grand public, afin de lui permettre de mieux saisir les causes des évolutions actuelles et apprécier leurs effets positifs ou négatifs. J'ai insisté sur deux sujets essentiels pour l'avenir : la sûreté des logiciels, trop souvent atteints par des bugs allant du pénible au dangereux, et la sécurité informatique, qui devient partout un problème majeur à cause de la multiplication et de l'industrialisation des vols de données et des attaques contre les systèmes informatisés (transports, hôpitaux, énergie, industrie, etc.). Enfin, j'ai aussi insisté sur l'importance de l'éducation à l'informatique, qui se met enfin en place au lycée. Les séminaires ont approfondi certains de ces points.

Cours 1 – Où va l'informatique ?

J'ai choisi de d'abord présenter le sujet global dans toute sa largeur dans un cours double, donc sans séminaire. Dans une première partie, j'ai rappelé que l'informatique est fondée sur quatre piliers : les données, les algorithmes, les langages et les machines, accompagnés des interfaces avec les hommes ou les autres machines. J'ai ensuite expliqué pourquoi l'information qu'elle manipule est très différente des objets d'étude des sciences classiques que sont la matière, l'énergie et les ondes, et pourquoi son traitement donne de nouveaux leviers d'une puissance considérable. J'ai illustré ce point essentiel par des exemples de traitement d'images photographiques ou médicales, puis d'informatisation de l'instrumentation scientifique et de généralisation de la simulation numérique. J'ai ensuite rappelé l'importance de la loi de Moore (doublement tous les deux ans du nombre de transistors par centimètre carré de silicium), qui a conduit l'évolution de l'infrastructure matérielle en permettant le passage des grands systèmes du passé à des objets portables bien plus puissants comme les smartphones, qui ont rendu l'informatique ubiquitaire, ainsi que l'évolution de l'infrastructure logicielle associée avec la généralisation de la mise en réseau des machines, du Web et du calcul dans le nuage (*cloud computing*). J'ai enfin souligné deux tendances plus récentes : le recentrage sur les données, dû aux nouvelles possibilités d'analyse permises par des algorithmes probabilistes, statistiques et d'apprentissage automatique, et l'essor de l'informatisation généralisée des objets physiques de tous les jours qui conduit à ce qu'on appelle l'IoT (*Internet of Things*).

Dans la seconde partie, j'ai d'abord analysé le nouveau paysage des applications informatiques, qu'elles soient réalisées par des sociétés privées devenues des acteurs économiques primordiaux ou par de grandes communautés de développeurs et d'utilisateurs comme les grands sites participatifs (Wikipedia), les grands logiciels libres (Linux), ou les nouvelles aventures de préservation du paysage comme Software Heritage. J'ai ensuite abordé deux des grands dangers de l'informatique : les problèmes de sûreté, avec plusieurs exemples de désastres dus aux bugs informatiques (en automobile, spatial, etc.), et les problèmes de sécurité, auquel le cours 3 ci-dessous a été consacré. Enfin, pour traiter plus spécifiquement l'évolution d'un domaine particulier, j'ai discuté les évolutions actuelles de l'informatique médicale, qui sont rapides mais demandent une adaptation des différents acteurs médicaux et informatiques qui sera loin d'être simple.

Cours 2 – Enseigner l'informatique

Le système d'enseignement primaire français a longtemps choisi d'ignorer l'informatique en tant que matière à enseigner, la confondant avec la compétence dans l'utilisation des ordinateurs – ce qui n'a pas grand-chose à voir. Elle n'est matière à part entière au lycée que depuis 2019, avec un cours général en seconde et une option du baccalauréat en première et terminale. La situation est assez diverse selon les pays. Par exemple, en Pologne, elle est enseignée depuis longtemps, en Angleterre, elle a été déclarée matière à part entière dès 2010, aux États-Unis, le président Obama s'est engagé personnellement pour son enseignement, et elle est vue comme une matière de premier plan en Inde ou en Corée du Sud. Après avoir rappelé l'histoire française de son enseignement et la situation des autres pays d'Europe, le cours a établi une distinction entre deux niveaux d'enseignement : la littéracie numérique, et la vraie compréhension des concepts centraux indispensables pour pouvoir comprendre ou créer de nouvelles applications dans n'importe quel domaine. Il a insisté sur l'importance de l'enseignement en primaire et au Collège, hélas encore bien timide, qui peut avantageusement passer par un enseignement et des exercices débranchés (*unplugged*) permettant de bien comprendre les principaux concepts sans utiliser de dispositif informatique. Il a enfin détaillé les nouveaux programmes du lycée dont j'ai été copilote de la conception et de l'écriture au CSP (Conseil supérieur des programmes) et les raisons des choix qui y ont été effectués.

Séminaire 2 – Automatisation de la maintenance du noyau de Linux : l'expérience Coccinelle

Julia Lawall (directrice de recherche à l'Inria Paris)

Le noyau du système d'exploitation Linux réalise l'ensemble des interfaces entre le matériel et les logiciels d'application. C'est un logiciel très gros (13 millions de lignes de code) et très complexe, écrit par plusieurs milliers de contributeurs. Il est développé de façon modulaire, les modules du cœur et les pilotes de périphériques (*drivers*) interagissant à travers des API (*Application Programming Interfaces*) qui spécifient comment leur parler. Ce noyau évolue en permanence, avec de d'ordre de 15 000 modifications par trimestre, les API étant souvent modifiées pour corriger des erreurs ou introduire de nouvelles fonctionnalités. Chacune de ces modifications d'API entraîne de nombreuses modifications dans de nombreuses parties du code, qu'il est très pénible à faire à la main et à valider. Fondé sur des analyses automatiques

fines de la sémantique des codes concernés, le logiciel Coccinelle développé par Julia Lawall et Gilles Muller permet d'automatiser presque complètement la mise à jour du système. Il est maintenant utilisé à grande échelle pour le développement de Linux.

Cours 3 – Les aspects scientifiques de la sécurité informatique

La sécurité informatique devient un problème crucial dans tous les pays et toutes les activités informatisées privées ou industrielles, au point que les attaques sur les données et systèmes sont classées comme dangers économiques majeurs par le *World Economic Forum*, juste derrière les catastrophes climatiques et les désastres naturels. Le cours a d'abord analysé les comportements des divers acteurs : personnes instruites ou non, DSI (directeur des services informatiques), dirigeants, fabricants de logiciels et d'objets connectés, tous trop souvent ignorants des questions de sécurité informatique, et bien sûr attaquants, eux très compétents et bien organisés. Il a ensuite présenté un florilège d'attaques récentes de grande envergure, vols massifs de données d'abord, ce dont on parle surtout dans la presse, mais aussi intrusions dans les grands systèmes informatisés (réseaux d'énergie, hôpitaux, industries) et dans les objets connectés (automobiles, *pacemakers*, etc.). Ces dernières attaques sont particulièrement dangereuses mais restent largement ignorées du grand public. Le cours a ensuite analysé le fonctionnement des attaques, soit par exploitation de faiblesses ou d'incuries des utilisateurs, soit par exploitation dans les systèmes informatisés de bugs souvent minimes ou de canaux cachés d'obtention d'information. Il a ensuite analysé les apports de la recherche en montrant comment des recherches en théorie des nombres et des graphes ou en logique mathématique mécanisée permettent d'augmenter, voire de prouver la sécurité de parties critiques des systèmes : méthodes de chiffrement efficaces et sûres, protocoles d'échanges de clef de chiffrement, vote électronique, etc. Le cours s'est terminé par l'analyse de failles sournoises trouvées récemment à l'intérieur même des microprocesseurs (*Meltdown*, *Spectre*, etc.).

Séminaire 3 – Sécurité numérique : sommes-nous condamnés à une lutte inégale entre le glaive et le bouclier ?

Guillaume Poupard (directeur général de l'ANSSI, Agence nationale de la sécurité des systèmes informatiques)

L'ANSSI est l'organisme national chargé de la sécurité des systèmes informatiques au sein de l'état, mais aussi d'organisation ou d'entreprises pour lesquels les questions de sécurité peuvent être critiques. Le séminaire a décrit les deux types d'attaques qui se généralisent de plus en plus, sur les données et sur les systèmes informatisés, leurs effets, la façon dont elles sont conduites, et leurs auteurs potentiels dont l'identité reste toujours difficile à prouver. Il a insisté sur la l'impréparation des acteurs dans beaucoup de domaines, par exemple avec la généralisation des petits automates programmables et des objets connectés peu sécurisés, l'utilisation restant fréquente de systèmes obsolètes comme Windows XP, le danger des mises à jour non faites à temps, ou la mise en place de machines à voter mal conçues. Il a noté l'importance des effets indirects comme la perte de confiance dans des acteurs essentiels en cas d'attaque. Il a détaillé tous les apports de la science au sujet, avec les progrès majeurs en cryptographie bien sûr, mais aussi le

développement de logiciels durcis comme une version de Linux développée à l'ANSSI. Il a enfin insisté sur le fait que les problèmes de sécurité sont solubles à condition qu'on les prenne au bon niveau, très élevé, que la conception des programmes et systèmes doit intégrer la sécurité dès le début du design et tout au long du cycle de vie des produits, et que deux questions cruciales restent la formation de tous les acteurs et la construction d'une législation nationale et internationale appropriée, dont la RGPD est un bon début.

Cours 4 – Compléments et réponses aux questions

Ce cours d'un format un peu différent a été consacré à la réponse à des questions envoyées auparavant par des auditeurs. Il a abordé six points principaux : l'importance cruciale des interfaces et interactions homme-machine, souvent sous-estimée ; l'importance croissante des données massives et de leur traitement par les algorithmes d'apprentissage profond de la communauté d'intelligence artificielle ; le rapport entre connaissance descriptive et connaissance opératoire, qui rejoint le rapport entre théorie et pratique ; l'évolution des langages de programmation, avec d'autres façon de penser les programmes (programmation « chimique » par exemple) ; les problèmes posés aux professeurs et aux institutions par les nouveaux enseignements d'informatique aux jeunes enfants, collégiens et lycéens ; enfin, certaines questions économiques et sociales posées par les nouvelles façons de penser et de travailler introduites par l'informatique ou concernant la liaison recherche-industrie.

Séminaire 4 – Les enjeux de la recherche en informatique

Antoine Petit (président directeur général du CNRS)

Antoine Petit connaît bien l'informatique pour avoir été chercheur dans le domaine et avoir présidé l'Inria avant de prendre la présidence du CNRS. Il a d'abord présenté les débuts de la recherche avec les pionniers des années 1960-1970, l'informatique étant alors rejetée par les mathématiciens comme n'étant « pas une science », puis la création de l'Iria en 1967 sous l'impulsion de J.-J. Lions et la création de quelques labos universitaires pionniers au début des années 1970. S'en est suivie une période d'expansion de 1975 à 2000, avec la création de nombreux labos dans toute la France et l'expansion continue de l'Inria. Enfin, à partir de 2000, la consécration d'un domaine parvenu à maturité, avec la création de l'INS2I au CNRS en 2009 et l'extension à d'autres sciences, en particulier celles de la vie. La situation française actuelle se caractérise par la présence de beaucoup d'acteurs, une place honorable sur la scène internationale (par exemple dans les ERC), mais une présence hétérogène sur les sujets les plus « chauds ». Pour le futur, huit enjeux ont été analysés : l'enjeu de formation de la société en général, l'enjeu de l'ouverture avec l'essor de la science et des développements collaboratifs, l'enjeu de genre, les femmes étant très sous-représentées dans le domaine, l'enjeu de concurrence avec les autres pays, les enjeux de relations avec les grands groupes et les start-ups, celui des données massives, en particulier dans les sciences, et enfin celui des SHS en termes d'éthique, d'accessibilité et de confiance pour ce qui concerne l'informatisation grandissante de la société et de sciences.

Cours 5 – Leçon de clôture : plaidoyer pour les trajectoires non linéaires

La leçon de clôture de ma chaire, plus festive, a été l'occasion de rappeler les principaux jalons de ma carrière de chercheur depuis 1970 : langage TIF de traitement et interrogation de fichiers (1970-1973), inversion des calculs des programmes récursifs (1973-1976), étude mathématique des propriétés syntaxiques et sémantiques du λ-calcul et en particulier de ses propriétés de stabilité et séquentialité (1975-1982), théorie des automates (1989), modèles asynchrones, synchrones et vibratoires du parallélisme et langages de programmation associés (1983-), programmation temps réel en langages synchrones (1983-2009), conception à haut niveau de circuits digitaux et caractérisation logique de leur bon fonctionnement (1989-2009), vérification formelle de programmes et circuit (1991-2009), et programmation réactive sur le Web et en musique depuis 2013. Tout ceci s'est fait avec pas mal d'incursions dans des sujets latéraux et un passage dans l'industrie comme directeur scientifique de la société Esterel Technologies dédiée aux applications industrielles de mon langage Esterel et d'autres langages synchrones (2001-2009). Ma trajectoire n'a donc pas été linéaire, même si j'ai toujours gardé comme ligne de force le besoin de soigneusement harmoniser syntaxe et sémantique mathématique dans les modèles et langages de programmation. Elle a varié en fonction de l'évolution des sujets et de leur impact sur le monde, et surtout en fonction de nombreuses rencontres de chercheurs passionnants ayant des façons de penser différentes, ce qui m'a permis d'avoir des idées théoriques non classiques que j'ai pu pousser au bout dans la pratique pour certaines. Pour moi, les trajectoires non linéaires peuvent être plus fécondes en recherche que les trajectoires linéaires, au moins dans un sujet en plein développement comme la science informatique.

J'ai terminé cette dernière leçon par quelques réflexions sur l'enseignement et un grand remerciement aux 66 séminaristes que j'ai eu l'occasion de faire intervenir dans mes cours et aux 11 titulaires successifs de la chaire Informatique et sciences numériques que j'avais inaugurée en 2009-2010, et bien sûr au Collège de France lui-même, car y enseigner a été un immense honneur.

Colloque – L'imagerie médicale à l'heure de l'IA ?

Comme celui de l'année précédente, ce colloque a été dédié aux algorithmes en médecine, et en particulier aux impacts des algorithmes d'apprentissage profond développés par la communauté des chercheurs en intelligence artificielle dans le domaine de l'imagerie médicale.

Enseignement à l'extérieur : Bordeaux, Inria

Cours : « En théorie, la théorie et la pratique, c'est pareil ; en pratique, ce n'est pas vrai » (le 12 décembre 2018)

Le rapport entre théorie et pratique n'est jamais simple en sciences, et l'informatique n'échappe pas à la règle. Le cours a illustré ce point dans plusieurs domaines. D'abord, le fait que les probabilités qui sont à la base de beaucoup d'algorithmes récents sont tout sauf intuitives, comme le montre le célèbre exemple du problème bien connu des trois portes (*Monty Hall Problem*) qui a défrayé la

chronique dans les années 1990, beaucoup de scientifiques s'accrochant à des raisonnements incorrects. Ensuite, la différence entre complexité théorique et pratique des algorithmes, à travers le problème SAT de satisfaction booléenne déjà étudié dans le cours 2015-2016 : si sa complexité théorique est supposée exponentielle dans les pires cas, ce qui a longtemps freiné la recherche d'algorithmes efficaces, les algorithmes modernes sont de fait très efficaces pour des problèmes de grande importance pratique, sans qu'on sache vraiment pourquoi. Puis la preuve mathématique en machine de théorèmes, vue ici sous le prisme de l'utilisation de SAT pour découvrir et prouver de nouveaux résultats en théorie des nombres qui sont encore inaccessibles aux méthodes théoriques classiques. Ensuite encore, la généralisation des algorithmes randomisés efficaces en pratique, qui sont très différents des algorithmes déterministes classiques. Enfin, un retour sur l'informatisation des sciences, où l'informatique a été longtemps vue comme un simple outil technique alors qu'elle modifie en fait partout au plus haut point les méthodes de travail et d'attaque des problèmes.

Séminaire lié : « Coq : aspects pratiques de la théorie des types »

Yves Bertot (directeur de recherche à l'Inria Sophia-Antipolis)

Les assistants de preuve ont deux types d'applications : la construction de preuves formelles de correction de programmes ou d'algorithmes très complexes (compilateurs et analyseurs statiques, algorithmes distribués, protocoles cryptographiques, etc.), et celle de preuves formelles de théorèmes mathématiques requérant l'étude de nombreux cas complexes (théorèmes des quatre couleurs, théorème de Feit-Thompson sur la classification des groupes, conjecture de Kepler sur le rangement optimal des sphère, etc.). Ces preuves sont en général très volumineuses, ce qui pose deux problèmes majeurs. D'abord, aucun être humain ne peut en écrire tous les détails ; avec Coq, ceux-ci sont créés en machine suivant les stratégies et indications fines fournies par l'utilisateur. Ensuite, la confiance en la correction d'une preuve écrite à l'aide d'un assistant de preuve ne peut provenir que de celle qu'on accorde au programme de vérification de preuves présent dans le noyau de cet assistant.

À l'aide de quelques exemples, le séminaire a abordé les questions suivantes : comment diriger un assistant de preuves de façon à vérifier un programme ou un théorème mathématique ? Comment convaincre une tierce personne qu'un long développement en Coq prouve bien le résultat annoncé ? La réponse tient en deux aspects : en pratique, la construction de « patrons de preuve » similaires aux « patrons de conception » de la programmation ; en théorie, la compréhension de la logique mathématique utilisée par l'assistant de preuve, clef pour la validation de la correction de l'algorithme de vérification des preuves.

Recherche

Ma recherche a principalement concerné la poursuite du travail sur le langage HipHop, version d'Esterel adaptée au langage JavaScript, ce dernier étant de plus en plus l'outil standard pour le développement d'applications web. J'ai également continué à encadrer l'activité d'Alan Aboudib (ATER attaché à ma chaire) sur l'utilisation de la focalisation sur une petite zone (fovea) dans l'analyse d'images par apprentissage profond. Cette recherche n'a pas abouti à des publications, mais à

l'embauche d'Alan Aboudib comme directeur technique d'une jeune pousse du domaine. J'ai, enfin, continué ma coopération scientifique avec des musiciens et des acteurs de l'informatique musicale, en particulier à travers mon rôle de président du Conseil scientifique de l'Ircam.

En ce qui concerne HipHop, que je développe avec Manuel Serrano (directeur de recherches à l'Inria Sophia-Antipolis), j'ai poursuivi l'action dans le domaine médical entamée en 2018 avec Steven Belknap, professeur de médecine à la Northwestern University Chicago. Il s'agit de l'utilisation d'Esterel et HipHop pour la rédaction précise et la traçabilité automatique des protocoles médicaux d'administration de médicaments. Leur définition actuelle en langage naturel trop flou et leur suivi approximatif provoquent des dizaines de milliers d'erreurs médicales par an dans les hôpitaux aux États-Unis. À l'aide un premier exemple soumis à publication, nous montrons que HipHop est bien adapté à l'écriture rigoureuse et au suivi de ces protocoles, à travers par exemple des applications web ou des boîtes à pilules automatisées. Une telle automatisation pourrait aussi ouvrir la voie vers une analyse automatique des erreurs et de leurs causes, ainsi que des bénéfices des méthodes automatisées lorsque suffisamment de données seront récoltées par la traçabilité automatique des applications des protocoles.

J'ai également donné 30 conférences dans des congrès scientifiques, médicaux ou de juristes, à la Commission européenne, dans des lycées et dans diverses organisations ou associations destinées au grand public. J'ai, de plus, enregistré une conférence, « la pensée informatique », chez De Vive Voix, avec l'Académie des sciences, parue en disque compact ; son texte a été édité en livre par CNRS Éditions. Ces actions constituent pour moi une activité de recherche pédagogique essentielle, tant la connaissance de l'informatique moderne reste encore bien trop faible dans notre pays. Enfin, toujours sur la pédagogie, j'ai été copilote de la définition des programmes des nouvelles options informatiques pour les classes de première et de terminale.

PUBLICATIONS

BERRY G., *La Pensée informatique*, Paris, CNRS Éditions / De vive voix, coll. « Les grandes voix de la recherche », 2019.

RIEG L. et BERRY G., « Towards Coq-verified Esterel semantics and compiling », 2019, arXiv : abs/1909.12582 (soumis pour publication).

BERRY G. et SERRANO M., « HipHop.js: (A)Synchronous reactive web programming », 2019, soumis pour publication.