CONSEIL SCIENTIFIQUE DE L'INSTITUT DES SCIENCES DE L'INFORMATION ET LEURS INTERACTIONS

Pascal WEIL (président) ; Clémence MAGNIEN (secrétaire scientifique) ; Elsa ANGELINI ; Christian BARILLOT ; Laure BLANC-FÉRAUD ; Christine COLLET ; Hubert COMON-LUNDH ; Gérald CONREUR ; Pierre-Louis CURIEN ; Isabelle DEBLED-RENNESSON ; Olivier GASCUEL ; Patrick GIRARD ; Jean-Paul LAUMOND ; Anne-Catherine LETOURNEL ; Adeline NAZARENKO ; Laurence NIGAY ; Anibal OLLERO ; Henri PRADE ; Jean-François RASKIN ; Michel RAYNAL ; Michel RIVEILL ; Dominique ROSSIN ; Michèle SEBAG ; Serge TORRES.

Introduction

Les sciences de l'information sont aujourd'hui au cœur d'une révolution qui vient transformer l'activité industrielle et économique, le développement des sciences, les mécanismes sociaux et les comportements individuels. C'est un privilège pour la communauté des chercheurs de notre domaine et aussi une responsabilité. Il nous faut participer au développement rapide de notre discipline tout en répondant aux sollicitations et aux questionnements passionnants qui émanent de notre environnement scientifique, économique et social.

Ces sollicitations et questionnements incluent à l'évidence de très nombreux défis scientifiques et technologiques ; mais aussi la lente prise en compte du statut des (relativement nouvelles) sciences du calcul et traitement de l'information dans le champ de la connaissance et de leur influence sur la façon même de pratiquer l'activité scientifique ; et bien sûr des questionnements sur l'impact rapidement croissant de ces sciences sur les transformations de notre monde et de nos sociétés.

C'est donc bien des *sciences de l'information et de leurs interactions* qu'il sera question ici.

Ce rapport, rédigé à l'été 2014, vient après de nombreux autres rapports solides et documentés, issus d'institutions françaises et étrangères, auxquels nous renvoyons volontiers nos lecteurs. On citera par exemple :
- la Stratégie Nationale de la Recherche, et en particulier le rapport de l'atelier *Société de l'information et de la communication*,[1]
- la présentation de l'Alliance Allistène et ses contributions à la Stratégie Nationale de la Recherche,[2]
- le Plan Stratégique 2013-2017 d'Inria,[3]
- le Rapport Lauvergeon,[4]
- le programme Horizon 2020 de l'UE,[5]
- les programmes des agences de recherche des pays européens, comme par exemple l'EPSRC au Royaume-Uni,[6]

mais aussi des rapports consacrés à des champs plus spécialisés au sein de nos disciplines :
- *Big Data across the Federal Government*, (White House, 2012),[7]

[1] http://www.enseignementsup-recherche.gouv.fr/cid78802/strategie-nationale-de-recherche-bilan-des-travaux-des-10-ateliers.html
[2] www.allistene.fr
[3] www.inria.fr/institut/strategie/plan-strategique
[4] innovation-2030.entreprises.gouv.fr/
[5] ec.europa.eu/programmes/horizon2020/en/what-horizon-2020
[6] www.epsrc.ac.uk
[7] http://www.whitehouse.gov/sites/default/files/microsites/ostp/big_data_fact_sheet_final.pdf

- *Synergistic Challenges in Data-Intensive Science and Exascale Computing*, (Department of Energy, mars 2013),[1]

- 2020 Sciences, par Microsoft.[2]

Beaucoup de ces rapports sont organisés en fonction des grands défis sociétaux auxquels les sciences de l'information peuvent contribuer : énergie, renouveau industriel, santé, mobilité et systèmes urbains durables, évolutions sociales et nouveaux modes de gouvernance, etc. Dans le même temps, tous insistent sur la nécessité de ne pas se focaliser sur les applications d'aujourd'hui mais de préparer celles de demain, alors même qu'elles sont encore inconnues mais qu'on pressent bien qu'en découlera notre capacité à développer de façon durable et soutenable nos économies et nos sociétés. Bref, si la poursuite d'une réflexion en profondeur sur le long terme et d'une recherche fondamentale apparaît indispensable aux yeux des rédacteurs de ces rapports, elle semble d'abord motivée par la réponse aux enjeux sociétaux. Nous avons souhaité prendre une perspective différente et partir de la notion de *défis scientifiques*, indépendamment de leurs applications.

La liste des défis passionnants auxquels nous pouvions nous attacher, et auxquels se consacrent déjà des équipes en France et dans le monde, est très grande. Certains résistent depuis longtemps et n'en sont que plus dignes d'intérêt et plus intrigants. Ils préoccupent nos communautés depuis de nombreuses années et sont très bien identifiés dans la plupart des rapports ; ils ne seront pas repris ici. Nous avons fait un choix éditorial qui met l'accent sur l'identification de sujets « nouveaux », dont *l'énoncé même aurait été difficile ou du moins très différent il y a cinq ou dix ans*. Ce choix est évidemment subjectif et nous assumons cette subjectivité.

Notre second choix éditorial concerne bien sûr la sélection des sujets retenus : nous avons opté pour la discussion d'un relativement petit nombre de défis, sans aucune prétention à l'exhaustivité ou à l'homogénéité. D'autres défis auraient pu être développés dans ce rapport, certains de ceux que nous avons retenus couvrent des champs assez larges alors que d'autres sont plus focalisés. Nous espérons que cette diversité des questionnements et des points de vue permet de percevoir la richesse et le dynamisme actuels des sciences de l'information.

L'élaboration des textes qui suivent a fait l'objet de longues discussions au sein du Conseil Scientifique de l'INS2I, dont nous souhaitons extraire dès maintenant quelques remarques.

- Comme tous les champs scientifiques, les sciences de l'information sont subdivisées en différentes disciplines qui ont chacune leur histoire, leur culture, leurs problématiques et leurs interactions avec des secteurs du monde socio-économique et avec les autres sciences, bref leur existence et leur légitimité. Nous observons cependant avec plaisir que la plupart des défis que nous présentons traversent ces frontières disciplinaires et contribuent ainsi à mieux identifier la spécificité des sciences de l'information.
- Il est très important de continuer à maintenir un équilibre – dans le fonctionnement de l'INS2I, mais plus généralement du CNRS et de la recherche publique – entre le soutien apporté à la recherche sur les fondements disciplinaires, à la recherche pluri ou interdisciplinaire, et à la réponse aux grands problèmes économiques et sociétaux. Les textes qui suivent, centrés sur des problématiques d'abord scientifiques, constituent de ce point de vue, selon les cas, un complément ou un contrepoint que nous espérons utile aux rapports mentionnés plus haut.
- Les solutions proposées pour ces défis scientifiques doivent s'accompagner d'une étude de leur impact sur la société, et les droits individuels et collectifs.

Une dernière remarque en forme de regret pour conclure cette introduction : nous aurions aimé labourer davantage le champ très riche des défis scientifiques partagés avec les autres instituts, particulièrement ceux avec lesquels nos laboratoires entretiennent le plus de collaborations : l'INSIS, l'INSB, l'INSMI et l'INSHS.

1 http://www.sci.utah.edu/publications/chen13/ASCAC_Data_Intensive_Computing_report_final.pdf

2 http://research.microsoft.com/en-us/um/cambridge/projects/towards2020science

I. Science des données : le tout n'est pas la somme des parties

La science des données fait intervenir plusieurs tâches (gestion, traitement, visualisation), plusieurs perspectives (celles des experts du domaine, du traitement de l'information et des statistiques, de l'exploitation des connaissances, de la gestion des données, du calcul haute performance), et plusieurs types de représentations (logiques, structurées, probabilistes, possibilistes, annotées).

L'état de l'art aborde de manière indépendante les diverses étapes du processus, relatives à la gestion des données et au traitement de l'information. Le problème est qu'une succession de décisions localement optimales ne définit pas une stratégie optimale. Ce problème constitue une limitation centrale pour la science des grandes données (*SgD*). Un objectif scientifique pour cette science consiste ainsi à définir la notion de stratégie SgD optimale ; un second, à la calculer. Le verrou réside dans le fait que le traitement et l'exploitation de grandes masses de données est un processus itératif et continu, dont le but est souvent connu *a posteriori*.

Le défi consiste à définir une approche intégrée pour modéliser l'ensemble des décisions et des critères en jeu dans un problème de SgD. Le but est d'aboutir à une stratégie optimale, adaptative en fonction du contexte et des experts, capable d'apprentissage au long cours (*lifelong learning*). Une stratégie correspond à une politique de traitement des données (PolD), qui sera vue comme un processus abstrait dans la suite du texte.

A. Etat de l'art

La science des grandes données se définit à l'intersection des masses de données, de l'apprentissage statistique, de la visualisation et de l'interaction homme-machine, du calcul haute performance ainsi que des sciences ou industries du domaine auxquelles appartiennent les données. Elle comprend l'ensemble des approches mathématiques, statistiques et informatiques utilisées pour extraire des informations, des

connaissances et des modèles à partir des données dans les domaines des sciences naturelles, humaines ou sociales (physique, sciences de l'univers, sciences de l'environnement, biologie, médecine, économie, sociologie, histoire, etc.), de l'industrie et du commerce (production, maintenance, marketing, veille, recommandation, assurances) et plus généralement dans le cadre d'un monde intégré (Internet et réseaux sociaux).

Les enjeux en sont scientifiques, économiques et sociétaux : la SgD est une source potentielle de bienfaits (en particulier pour la santé et l'environnement) et de menaces (en particulier pour la vie privée) ; ses modalités d'usage posent des questions scientifiques et éthiques fondamentales (analyse de causalité, traçabilité des décisions) que la recherche en informatique doit prendre en compte.

Les verrous majeurs sont scientifiques et technologiques, allant de la définition des objectifs aux procédés de collecte et de traitement des données, à l'analyse et la validation des résultats obtenus. Fondamentalement, la SgD implique un processus itératif en plusieurs étapes : définition de nouveaux objectifs en fonction des résultats ; reconsidération des algorithmes et des contextes, visant le passage à l'échelle ; reconsidération des formalismes de représentation, permettant d'accommoder les aspects structurés, annotés, probabilistes, contraints, possibilistes des données ; reconsidération des données (nouvelles acquisitions, prise en compte de la qualité des données) et des connaissances du domaine pour remédier aux défauts et incomplétudes des résultats courants.

B. Défi

Le défi consiste à situer ces étapes dans une perspective conceptuelle, formelle, algorithmique et technologique intégrée : les décisions effectuées au coup par coup dans un processus de traitement et d'exploitation de grandes masses de données ne sont pas indépendantes. La vision proposée est celle d'un problème de décision séquentielle, où les décisions locales doivent être prises en tenant compte d'un objectif global partiellement connu au moment où la décision locale est prise. De plus, le problème de décision posé est généralement multi-critères, en raison notamment de sa nature pluri-disciplinaire.

Il s'agit donc de munir le processus PolD d'une mémoire et de critères internes visant l'exécution cohérente des différentes étapes d'une session de traitement, et l'amélioration des performances au cours des sessions.

La suite de cette section décrit quatre axes de recherche possibles pour PolD, puis leur instanciation pour deux domaines d'applications, les sciences du vivant et les sciences humaines et sociales.

1. Représentations

Le bien-fondé des résultats obtenus par composition de tâches demande une prise en compte unifiée de formalismes de représentation des données et connaissances (incluant leurs dimensions d'hétérogénéité, de qualité, et leur mode de production) et de leurs annotations (probabilistes ou non), d'un bout à l'autre de la chaîne des traitements. Il s'agit en particulier d'assurer la cohérence des modèles considérés et des hypothèses statistiques sous-jacentes lors des diverses étapes.

L'analyse de la qualité des données et de leurs origines, les transformations qu'elles ont subies et leur durée de vie sont également des dimensions de nature à influer sur leur interprétation.

2. Qualité des données et connaissances *a priori*

La collecte et le nettoyage des données sont notoirement des tâches cruciales et fastidieuses, demandant jusqu'à 90 % du temps des experts pour une application donnée. La question consiste à savoir si et comment ces deux étapes peuvent être abordées à haut niveau.

Un jalon pourra s'intéresser aux notions de mixtures de données (les données reflétant souvent la superposition de données issues de phénomènes différents, par exemple dans le cas des bases de données d'électro-encéphalogrammes). Un autre jalon concerne le critère de confidentialité et le *reverse engineering* du processus PolD.

3. Langages, conception et passage à l'échelle

Les algorithmes doivent être pensés dès leur conception en termes des propriétés voulues (passage à l'échelle et portabilité). Ainsi, dans le domaine des bases de données probabilistes, les algorithmes de Chen et al. 2013 [1] et Suciu *et al*. 2011 [3], validés en C++, n'ont pas préservé leur niveau de performance quand ils ont été traduits dans des langages portables tels que Java ou Python.

Le fait de prendre en compte de manière intégrée les objectifs de qualité et de faisabilité (pertinence des résultats, passage à l'échelle) du PolD pourra passer par l'identification et la réalisation d'un ensemble de primitives de traitement qui implémente un langage de haut niveau d'exécution optimisée. Ce langage doit aller au delà par exemple du langage Pig Latin, qui certes réduit le temps de développement et facilite l'expression des tâches d'analyses sur Hadoop en permettant des sélections, projections, jointures, opérations d'agrégation de base sur de très grands ensembles de données, mais n'est pas suffisant pour l'expression de traitements complexes sur les données et pour la garantie d'une exécution efficace de ces opérations.

Un jalon possible concerne la recherche de plans d'exécution dynamiques, en supposant le cas de données intentionnelles (par ex. la collecte des données fait partie des opérations, et elle a un coût variable selon les sources et le type de données). Dans ce contexte, nous avons besoin de maintenir une évaluation avec intervalle de confiance des différents plans de requête possibles, de les réviser, et de faire en sorte que les plans et les résultats obtenus soient réutilisables d'une session à une autre.

4. Séquence de requêtes et connaissances du domaine

L'enjeu d'unifier les données et les connaissances *a priori* est de permettre à l'utilisateur « d'élever le niveau de sa demande », par exemple en remplaçant la séquence de requêtes *Quels sont les restaurants proches d'ici, quelles sont leurs évaluations, comment s'y rendre*, par : *Trouver un restaurant dans mes goûts proche d'ici* – comme dans Pang, Bo et Ravi Kumar [2]. Formellement, le cadre PolD doit permettre l'inférence sous toutes ses formes, déductive, inductive, abductive et analogique, en tenant compte des divers formalismes de représentation. L'ajout d'une fonctionnalité argumentative est approprié pour certains domaines.

C. Exemples d'utilisation de PolD

1. Le domaine du vivant

La modélisation du vivant soulève des défis sur plusieurs niveaux : la gestion de données hétérogènes, la modélisation multi-échelle (du nano au corps entier), l'intégration et réduction des modèles, le calcul à grande échelle, l'apprentissage, la confrontation aux données (image et physiologie), l'analyse quantitative, l'assimilation d'image. Ces défis se posent de façon nouvelle du fait des progrès toujours rapides des systèmes d'acquisition et d'analyse, particulièrement en biologie. Une conséquence est que les *représentations* évoluent elles aussi avec les capteurs d'imagerie (variabilité intrinsèque entre sujets, modèle de représentation d'apparence robuste pour les images, etc.). La *qualité des données et les connaissances a priori* jouent un rôle fondamental pour l'annotation de données numériques médicales par les experts, et cette annotation est nécessaire au développement de méthodes d'apprentissage qui permettront de découvrir de nouveaux modèles, d'exploiter plus finement la richesse de l'information fournie par les données numériques et ainsi d'aller au-delà de la simple réplique du travail de l'expert. Le *calcul haute performance* est naturellement un enjeu majeur du modèle numérique pour la santé : il faut gérer des énormes bases de données hétérogènes et complexes (images, génomique, etc.) et pouvoir rechercher des données par *séquences de requêtes*. L'investigation empirique est au cœur du développement de ces modèles, et requiert des outils de calculs rapides et flexibles tels que ceux proposés dans une plateforme PolD.

À ce jour, la médecine computationnelle a permis d'améliorer la compréhension de pathologies. Très peu de modèles sont

actuellement utilisés en routine clinique, et tous sont encore à un stade préliminaire de développement. L'usage généralisé de tels modèles requiert une validation rigoureuse, tant sous l'angle technique qu'en termes d'apports pour le patient et le système de santé. Ceci passe par l'intégration des différents acteurs dans des équipes uniques interdisciplinaires, regroupant physiciens, mathématiciens appliqués, biologistes, médecins, informaticiens, et par l'implication des industriels au niveau des équipes de recherche interdisciplinaires.

2. Le champ des sciences humaines et sociales

La SgD peut fournir aux SHS (droit, psychologie, géographie, économie, histoire, etc.) le micro/macroscope permettant de voir ce qui échappe aux méthodes d'analyse traditionnelles. Une meilleure compréhension et maîtrise de l'ensemble de la « chaîne de valeur » liée aux données doit permettre une meilleure acceptation sociétale des méthodes de la SgD et leur développement raisonné, au moment où la prise en compte de données massives amène des changements épistémologiques dans les SHS [4].

Comme dans d'autres domaines, le verrou vient de l'hétérogénéité et de la dispersion des données des SHS. Celles-ci travaillent quasi-exclusivement sur des données attestées mais les données recueillies sont multidimensionnelles (dimensions géographique, temporelle, sociologique, culturelle, etc.) et de natures diverses (textes, images, vidéos, etc.). Il s'agit à la fois de données primaires dont la numérisation est en soi problématique (données patrimoniales, par exemple) et de données secondaires issues d'un premier travail de sélection, filtrage, nettoyage, analyse, etc. Cette hétérogénéité fait que les données, aussi massives soient-elles, sont toujours bruitées, lacunaires et déséquilibrées ; ces biais doivent être pris en compte par un processus PoID.

La complexité des données ajoute une difficulté supplémentaire. Les données sont organisées (les mots d'un texte ne sont pas distribués au hasard, l'interprétation juridique raisonne sur les liens entre les textes de lois, décrets, jurisprudence, etc., l'analyse historique repose sur différentes échelles spatio-temporelles, etc.) et le PoID doit intégrer cette connaissance *a priori* : il faut pouvoir manipuler des structures variées (séquences, graphes, arbres ou hiérarchies) souvent hybrides.

La prise en compte des dimensions humaines et sociales des données nécessite aussi de pouvoir modéliser et exploiter leur dimension argumentative. On sait par exemple que les opinions, expertises, débats abondent sur le Web sur toutes sortes de sujets. Être capable, sur un sujet donné, d'analyser, d'évaluer, de synthétiser des points de vue exprimés, d'identifier les principaux arguments échangés et les points de désaccord, constitue un défi supplémentaire pour comprendre l'homme dans son environnement social.

Références

[1] Chen, Lei, Ihab F. Ilyas, Christopher Ré, and Xiaofang Zhou (2013). Probabilistic Web Data Management. In: World Wide Web 16.3, pp. 271-272.

[2] Pang, Bo and Ravi Kumar (2011). Search in the Lost Sense of Query: Question Formulation in Web Search Queries and its Temporal Changes. In: ACL (Short Papers). The Association for Computer Linguistics, pp. 135-140.

[3] Suciu, Dan, Dan Olteanu, Christopher Ré, and Christoph Koch (2011). Probabilistic Databases. Synthesis Lectures on Data Management. Morgan & Claypool Publishers.

[4] Williford Christa, and Charles Henry (2012). One Culture. Computationally Intensive Research in the Humanities and Social Sciences, Council on Libraries and Information Resources, pub151.

II. Systèmes centrés humains

A. Convergence du monde numérique et du monde physique

Dans le contexte de la convergence accélérée du monde numérique et du monde physique, il ne s'agit pas seulement d'exploiter au mieux les ressources techniques, mais aussi de développer des techniques d'interaction utiles, utilisables voire plaisantes, qui soient donc conformes aux attributs de l'interaction humaine et aux contextes d'interaction. Depuis l'interface graphique à manipulation directe du Xerox Star (1981), les travaux de recherche sur l'interaction homme-machine (IHM) sont centrés sur un même espace d'interaction confiné à l'ordinateur (de taille variable : ordinateur de bureau, portable et de poche) avec des dispositifs d'interaction comme un clavier-souris-écran. De l'essor technologique (comme la miniaturisation des composants électroniques et l'omniprésence des réseaux), nous retenons que l'espace d'interaction entre l'utilisateur et les données/traitements numériques devient plus vaste, il comprend l'environnement physique (*The shifting boundary between computers and the everyday world* [4]). Comme l'indique le rapport EPSRC [1], *The `C' in Human-Computer Interaction has changed radically, ... creating a physical-digital ecosystem*. L'objectif à long terme est de bénéficier des ressources informatiques, sans avoir d'une part à se couper du monde physique, ni d'autre part devoir se priver de nos moyens d'interaction avec le monde physique. Quelles sont alors les caractéristiques du monde physique qui ont un impact sur l'interaction homme-machine ? Comment influent-elles sur la conception et le développement logiciel des techniques d'interaction ?

Parallèlement, la convergence des mondes numérique et physique passe aussi par une plus grande autonomie et ubiquité des robots : en effet les robots sortent des sites industriels dans lesquels ils étaient jusqu'ici confinés pour partager le quotidien d'usagers non experts (les robots aspirateurs sont vendus par millions). L'homme fait de plus en plus partie intégrante de l'environnement dans lequel le robot évolue. Son autonomie dépend de sa capacité à prendre en compte l'activité humaine. Quels sont les facteurs humains qui déterminent leurs usages ? Quels en sont les fondements calculatoires qui peuvent permettre à une machine de les prendre en compte ? Comment impactent-ils les architectures logicielles, les lois de commande, la conception physique des systèmes ?

B. Modéliser les caractéristiques du monde physique et celles de l'homme et de ses fonctions sensori-motrices

Le verrou scientifique majeur lié à la fusion harmonieuse des mondes numérique et physique réside dans la modélisation des facteurs du monde physique et dans celle des facteurs humains, dans leurs composantes à la fois actionnelles, perceptuelles et cognitives, pour la conception de systèmes centrés humains.

Au sein d'une approche par induction partant de modèles fondamentaux pour concevoir et évaluer de nouvelles formes d'interaction, le défi consiste à prendre en compte les facteurs liés au monde physique, c'est-à-dire à identifier et organiser les variables de conception liées au monde physique. Ainsi les modèles prédictifs de performance [humain - technique d'interaction] doivent prendre en compte les variables physiques comme la forme ou la rugosité de la surface d'interaction lors d'un geste (la surface n'est plus nécessairement lisse et plane) ou encore la distance physique à l'objet d'interaction (modélisation de performance à partir de distances angulaires mais aussi prise en compte de la proxémique avec l'utilisation de l'espace physique). Dans un environnement physique interactif, peuplé d'objets physico-numériques, il faut également tenir compte, au delà des paramètres humains de type sensori-moteur ou perceptif, des paramètres cognitifs comme le coût du passage d'une technique d'interaction à une autre ou le coût d'apprentissage d'une technique (transfert de compétences du monde physique au monde numérique et vice-versa).

Au sein d'une approche par déduction partant de l'expérimentation de nouvelles techniques d'interaction pour enrichir des modèles fondamentaux, de nouvelles formes d'interaction doivent être conçues et développées. Au-delà des défis techniques (capteurs, effecteurs et matériaux intelligents

[3]), il faut se confronter aux défis conceptuels suivants.

- Les techniques d'interaction à concevoir concernent l'interaction homme-machine comme les interfaces tangibles et l'interaction homme-robot, ainsi que l'interaction homme-environnement comme l'interaction à distance. La conception de techniques d'interaction physique implique la prise en compte de nouveaux facteurs comme la « résolution de la forme » des dispositifs [6] (au même titre que la résolution d'un écran ou d'une souris).
- Le développement de techniques d'interaction implique d'étudier les paradigmes de programmation (comme l'expression du couplage de propriétés physiques d'un objet avec des variables numériques) et les outils logiciels de prototypage/développement : l'objectif visé est de faciliter le développement de techniques d'interaction dans le cadre d'une démarche expérimentale pour explorer l'espace de conception.

Le robot est une machine qui bouge. L'interaction physique entre l'homme et le robot pose en premier lieu un problème de sécurité qui se pose également de manière critique pour certains types de nouveaux robots, comme les robots volants. De nouveaux systèmes de motorisation, plus sûrs, imposent de nouveaux schémas de commande plus complexes, qui sont autant de défis pour l'algorithmique temps-réel, la commande des systèmes, le traitement du signal, la conception de nouveaux capteurs et systèmes de perception de l'environnement, et l'utilisation de nouveaux matériaux comme les alliages à mémoire de forme.

Un robot qui collabore avec l'homme doit traiter des modèles de mouvements et de comportements de l'homme. Modéliser l'homme dans cette perspective ouvre des champs scientifiques nouveaux tels que la recherche d'invariants sensori-moteurs dans la représentation de l'action humaine, et la modélisation de la dynamique des mouvements de l'homme en phase de collaboration avec le robot, champs qui s'abordent par la commande optimale inverse, une thématique très récente et difficile des mathématiques appliquées, en complément des approches par apprentissage.

L'objectif à terme, lorsque ces verrous scientifiques seront levés, est de rendre possible une interaction fluide et harmonieuse avec des éléments issus du monde physique et du monde numérique (*to systems that merge physical and virtual for both places and people* [5]).

C. Impact et positionnement

En partant du constat que tout système informatique est interactif, tous les secteurs socio-économiques sont concernés : environnement, alimentation, agriculture, communication, tourisme, habitat, transport (IHM voiture-avion), e-administration, santé, etc. Des exemples de marchés qui émergent incluent les objets interactifs et communicants dans la maison, les smartphones avancés et les objets dédiés portés sur soi (santé, sport).

Concernant la robotique, deux grands marchés sont en train de s'ouvrir avec la robotique de service (du robot guide dans les espaces publics au robot personnel à la maison, éventuellement intégrés avec les objets interactifs et communicants mentionnés plus haut) et la robotique collaborative en milieu industriel [2].

De nombreuses applications grand public sont à fortes retombées sociétales : elles concernent l'environnement interactif à la maison, le confort et le bien-être, le robot compagnon, etc.

Par la prise en compte des facteurs humains, le lien avec l'Institut des sciences humaines et sociales (INSHS) est évident et déjà ancien (psychologie, ethnographie, sociologie, ergonomie). La recherche en robotique entretient des liens endémiques avec l'INSIS dont elle est issue. Elle s'ouvre également depuis plusieurs années à la biomécanique et aux neurosciences (INSB). Des liens prometteurs pourraient s'établir avec l'Institut de Chimie (INC) et l'Institut de Physique (INP) pour tous les aspects liés aux sciences des matériaux (*smart materials*).

Références

[1] EPSRC March 2012. Report of the EPSRC Review of Human Computer Interaction Research in the UK. http://www.epsrc.ac.uk/SiteCollectionDocuments/Publications/reports/HCIReview.pdf

[2] H2020 Robotics 2020 Multi-Annual Roadmap http://www.eu-robotics.net/cms/upload/PDF/MultiAnnual_Roadmap_2020_Call_1_Initial_Release.pdf

[3] Ishii, H., Lakatos, D., Bonanni, L., Labrune, J.-B., 2012. Radical Atoms: Beyond Tangible Bits, Toward Trasnformable Materials. Interactions, 2012, 38-51.

[4] Microsoft 2008. Being Human. Human-Computer Interaction in the year 2020.

http://research.microsoft.com/en-us/um/cambridge/projects/hci2020/downloads/BeingHuman_A4.pdf
[5] NSF. Information and Intelligent Systems (IIS) http://www.nsf.gov/pubs/2013/nsf13580/nsf13580.pdf
[6] Roudaut, A., Karnik, A., Löchtefeld, M., Subramanian, S., 2013. Morphees: Toward High "Shape Resolution" in Self-Actuated Flexible Mobile Devices. ACM CHI 2013. 593-602.

III. Gestion de l'énergie et de la fiabilité : deux défis majeurs pour l'évolution des systèmes intégrés matériels-logiciels

Avec la microélectronique, nous avons assisté à un développement sans précédent du degré de miniaturisation, dans le domaine du traitement de l'information et de la communication. L'histoire des sciences met en évidence les besoins : automatiser, calculer, coder-décoder, gérer des données. Ces besoins ont suscité des progrès technologiques en mécanique, fluidique, puis électrique et électronique : depuis l'invention du tube à vide en 1904, du transistor en 1947, du premier circuit intégré en 1960, l'évolution des technologies de fabrication des circuits intégrés, permet de réaliser aujourd'hui des systèmes microélectroniques complexes intégrés sur une même puce : les SoC (pour System On Chip) sont des circuits d'une surface de l'ordre de quelques cm^2, intégrant plusieurs centaines de millions de transistors (par exemple 7 milliard de transistors pour le dernier FPGA de la famille Xilinx) ayant des longueurs de grille minimum de 28 à 22 nm en technologie CMOS. La complexité de ces circuits dépasse ainsi aujourd'hui largement le milliard de transistors, et ces technologies font évoluer nos besoins. Les enjeux sociaux et économiques sont en effet cruciaux du fait de la présence des technologies micro-nano-électroniques dans la totalité des équipements destinés aux technologies de l'information et de la communication. Les performances techniques recherchées pour les objets nomades communicants (téléphones mobiles, assistants numériques personnels, navigateurs GPS, etc.) sont une bonne illustration des objectifs à atteindre dans des marchés où la compétition internationale est très forte : faible poids, faible volume, grande autonomie, bonne couverture géographique, bonne ergonomie, performances permettant de transmettre en temps réel des informations audio ou vidéo éventuellement sécurisées, etc., et à faible coût. Par ailleurs, l'émergence de nouveaux usages et les projections d'une ère numérique « tout connecté » pose des problèmes nouveaux, marqués par des défis scientifiques dans un

contexte de prise de conscience de l'empreinte écologique.

A. Verrous scientifiques

Le domaine SoC-SiP, qui désigne les systèmes micro-nano-électroniques matériels-logiciels, intégrés en 2D et 3D, est donc fortement interdisciplinaire. La complexité des architectures permet d'envisager dans cette décennie d'embarquer plusieurs centaines de processeurs sur une même puce de silicium, avec une part prédominante du logiciel, ce qui pose de nouveaux défis liés à cette complexité (HPC, parallélisme, NoC ou réseaux sur puce, tolérances aux pannes, test, vérification, etc.), qui s'ajoutent à la prise en compte des limites technologiques. À cette dimension de complexité s'ajoute une problématique marquée d'hétérogénéité qui trouve son origine dans l'avènement de l'internet des objets. Le spectre de systèmes hôtes s'étale des serveurs de calcul (*cloud computing*) aux objets les plus anodins du quotidien de chacun tels que montres connectées ou vêtements intelligents (*wearable computing*), tous ayant vocation à être connectés. Les défis d'harmonisation, d'interopérabilité et de mise à l'échelle soulèvent bien sûr de nombreuses questions, avec des exigences en termes de stockage, de capacité d'indexation ou de puissance de calcul.

1. L'aspect énergétique

L'aspect énergétique est ici au cœur des préoccupations, pour des raisons financières (facture électrique) ou environnementales (empreinte carbone). À titre d'exemple, les machines les plus complexes (utilisées pour le *cloud computing* par exemple) offrent aujourd'hui des puissances de calcul dépassant le « pétaFLOPS », (soit 10^{15} ou un million de milliard d'opérations à virgule flottante par seconde), au prix d'une consommation énergétique de quelques mégawatts. L'énergie dépensée par ces centres de calcul et toute l'infrastructure distribuée autour du web et des communications mobiles (serveurs, réseaux, antennes, terminaux mobiles) constituera donc un des enjeux majeurs du développement durable. L'aspect énergétique est aussi au cœur des préoccupations pour des raisons fonctionnelles (autonomie des systèmes embarqués et/ou enfouis). Sur ce dernier point, la nécessité de trouver des solutions aux problèmes posés par la récupération d'énergie (*energy harvesting*) en vue d'une autonomie totale des systèmes intégrés matériels-logiciels soulève de nombreux défis. Il y a fort à parier que les gains exigés en efficacité énergétique passeront à moyen terme par l'exploitation de technologies alternatives.

2. La fiabilité

Par ailleurs, malgré les efforts réalisés ces dernières années pour améliorer la fiabilité des circuits et systèmes intégrés, l'évolution continue des technologies de semi-conducteurs (en termes de miniaturisation et de vitesse) les rend de plus en plus sensibles aux impacts environnementaux et agressions externes (radiations par exemple) et a tendance à réduire leur durée de vie. Comme dans le même temps l'utilisation de systèmes électroniques destinés à des applications critiques ne cesse de croître (dans l'automobile ou le domaine médical par exemple), il y a désormais une réelle nécessité à développer des techniques permettant de concevoir et de fabriquer des systèmes fiables à partir de « composants » (ou briques élémentaires) non fiables.

Ce défi peut être abordé en agissant à différents niveaux, du matériel au logiciel, en introduisant de la redondance pour détecter des fautes ou déviations de paramètres (tension ou fréquence par exemple) et tolérer ou masquer leurs effets.

En ce qui concerne le niveau d'abstraction matériel le plus élevé et à titre d'exemple, ce défi passe par la définition d'architectures de systèmes multiprocesseurs capables de faire face à ces problèmes de fiabilité par le biais de l'auto-adaptation, dont le principe réside dans la gestion autonome de la détection et de la correction en ligne des erreurs qui peuvent apparaître pendant le fonctionnement, et dans lequel un processeur « sain » pourra prendre le relai d'un processeur « défaillant » en cas de nécessité. Le poids des contraintes liées au surcoût en surface, à l'impact sur les performances, à la consommation supplémentaire, à la nécessité de travailler sur des architectures régulières, etc. représentera un défi majeur au développement rapide de solutions efficaces pour l'amélioration de la fiabilité des systèmes matériels-logiciels intelligents de demain.

B. Impact et positionnement

De très nombreux domaines mobilisent les systèmes intégrés matériels-logiciels : santé et aide à la personne, sécurité, transports terrestres et aériens, spatial, télécommunications, développement durable.

Par ailleurs, le développement de l'informatique (calculateur centralisé, PC distribués, accès au réseau, web, *cloud computing*) montre l'avènement des systèmes informatiques ubiquitaires qui permettent d'accéder à l'information sur tout type de terminaux mobiles communicants. Le *cloud computing* offre des services sur internet avec un accès partagé à un nombre « illimité » de ressources en réseaux de services, de calcul et de stockage. Ce déploiement du *cloud* est un exemple caractéristique des défis nouveaux à relever en termes d'énergie et de fiabilité des systèmes micro-nano-électroniques matériels-logiciels.

L'électronique est le point de départ et constitue le barycentre du domaine SoC-SiP, avec des interactions fortes d'un côté avec l'informatique, les mathématiques, le traitement du signal et de l'image, l'automatique, la robotique, et de l'autre la physique. Les défis mentionnés ci-dessus et liés à l'énergie et à la fiabilité se situent donc clairement à l'interface entre l'INS2I et l'INSIS, avec quelques interactions vers les autres instituts (mathématiques, physique, biologie, etc.)

IV. Programmation faible

Ce défi s'intéresse à la conception automatique de programmes par un humain non expert, ne pouvant ni spécifier ni démontrer le comportement attendu du programme.

A. Etat de l'art

Une méthodologie rigoureuse de conception et de certification de logiciels repose sur les notions de spécifications formelles et de preuve. Une alternative (mieux adaptée à des domaines liés à la langue naturelle et la recherche d'information, à la vision, ou à l'interaction avec un monde ouvert) se fonde sur l'apprentissage statistique supervisé à partir de données. L'apprentissage supervisé requiert une expertise forte à deux niveaux : i) le recueil des données; ii) le fait d'associer une valeur ou un sens à chaque donnée.

Le présent défi, appelé *Programmation Faible* (ProFab) s'intéresse à la conception de programmes par des utilisateurs d'expertise faible, ne pouvant ni spécifier ni démontrer le comportement attendu.

Ce défi s'inscrit dans une tendance émergente à la croisée de l'apprentissage statistique, de l'optimisation et de l'interaction homme-machine, dans le domaine du rendu visuel [2,3], de la recommandation [8], de la recherche d'information [7] et de la robotique [6,10,1].

ProFab vise une division du travail de programmation entre deux partenaires, un ordinateur limité en connaissances, et un utilisateur limité en capacités de calcul et en mémoire. Itérativement, le premier propose un comportement; le second renvoie un feedback de type « c'est mieux / c'est moins bien ».

B. Questions scientifiques

L'objectif diffère significativement de celui de l'interaction homme-machine (IHM). En IHM, la machine ne doit pas gêner le travail humain. Ici, le travail humain consiste précisément à interagir avec la machine et à l'éduquer. La machine a besoin d'une vérité de terrain (c'est mieux / c'est moins bien), parce que la réaction d'un humain en face d'un programme n'est pas

nécessairement corrélée à la qualité fonctionnelle du programme [7].

L'objectif de ProFab est de construire un programme, c'est-à-dire une fonction associant à tout état une action, telle que la suite de ces actions définisse un comportement approprié (voir les exemples ci-dessous). Il s'agit en tout cas bien ici de décision séquentielle (par ex. jouer au billard) et non d'apprentissage supervisé (par ex. savoir si une image contient ou non un visage).

Les verrous scientifiques sont les suivants. Premièrement, l'espace de recherche de la Programmation Faible doit être assez puissant et expressif pour représenter les comportements et les programmes voulus. En second lieu, cet espace doit permettre d'apprendre la fonction d'utilité de l'humain à partir d'un nombre faible d'interactions (quelques douzaines). Enfin, les préférences de l'humain évoluent en général au cours de l'interaction.

Les deux premiers verrous (un langage puissant mais permettant l'apprentissage) sont classiquement au cœur de l'apprentissage structuré depuis ses débuts. La difficulté additionnelle consiste ici à apprendre à partir d'une poignée d'exemples. Elle impose de disposer d'*a priori* efficaces pour canaliser la recherche de solutions. Le troisième verrou (l'objectif et les préférences changent en cours de route), qui augmente considérablement la difficulté de l'apprentissage statistique, peut être une solution aux deux premiers [9]. La succession des préférences peut implémenter les *a priori* efficaces : par exemple, l'humain peut vouloir apprendre à la machine à marcher, avant de lui apprendre à courir.

C. Impact et applications

En cas de succès, ce défi répond à la fracture numérique : un humain non informaticien, non expert du domaine d'application, et seulement capable de comparer deux comportements, obtient de la machine un comportement acceptable.

Ce défi répond au besoin d'une programmation *néoténique* : les conditions d'usage ne sont pas complètement connues au moment de la programmation, et la finalisation ou l'opérationalisation du programme s'effectue en situation. Cette évolution transpose dans le domaine logiciel les objectifs de l'*Autonomic Computing* dans le domaine computationnel et intergiciel [5].

Donnons deux exemples qui pourront servir de défis applicatifs.

Le premier concerne la décision subjective optimale. La fonction définissant la qualité d'un objet est subjective et non calculable. Par contre, l'humain peut évaluer immédiatement si la nouvelle solution proposée par la machine est en progrès par rapport à la dernière solution.

Le second concerne la décision séquentielle optimale. Dans le domaine de la robotique, l'approche par renforcement inverse a permis d'atteindre plus aisément le comportement désiré du robot en partant des démonstrations par l'expert de comportements (quasi) optimaux. Cette approche présente des limitations notamment parce que l'expert dispose d'un appareil sensoriel ou d'un moteur différent (comme des capteurs de retour d'effort). L'approche ProFab peut répondre à ces limitations. Un banc d'essai pourra concerner le fait pour un robot dépourvu de capteurs proprioceptifs d'attraper un verre en plastique.

Références

H2020 ICT 2014 - Information and Communications Technologies. Topic: *Multimodal and Natural computer interaction* ICT-22-2014.

ANR Plan d'action 2014, *Interactions des mondes physiques, de l'humain et du monde numérique*.

Microsoft: *The shifting boundary between computers and the everyday world*.

NSF Information and Intelligent Systems (IIS): *to systems that merge physical and virtual for both places and people*

[1] Akrour, R., Schoenauer, M., Sebag, M., and Souplet, J.-C. Programming by feedback, In *Int. Conf. on Machine Learning* (*ICML*), ACM Int. Conf. Proc. Series, 2014.

[2] Brochu, E., de Freitas, N., and Ghosh, A. Active preference learning with discrete choice data. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T. (eds.), *Neural Information Processing Systems*, 2007.

[3] Brochu, E., Brochu, T., and de Freitas, N. A Bayesian interactive optimization approach to procedural animation design. In Popovic, Z. and Otaduy, M. A. (eds.), *Symposium on Computer Animation*, pp. 103-112. Eurographics Association, 2010.

[4] Jain, A., Joachims, T., and Saxena, A. Learning trajectory preferences for manipulators via iterative improvement. In *Neural Information Processing Systems*, pp. 575-583, 2013.

[5] Kephart, Jeffrey O and Chess, David M. The vision of autonomic computing. *Computer*, 36: 0 41-50, 2003.

[6] Knox, W. B., Stone, P., and Breazeal, C. Training a robot via human feedback: A case study. In *Int. Conf. on Social Robotics*, volume 8239 of LNCS, pp. 460-470. Springer, 2013.

[7] Radlinski, F., Kurup, M., and Joachims, T. How does clickthrough data reflect retrieval quality? In J. G.

Shanahan *et al*. (ed.), *CIKM*, pp. 43-52. ACM Int. Conf. Proc. Series, 2008.
[8] Viappiani, P. and Boutilier, C. Optimal Bayesian recommendation sets and myopically optimal choice query sets. In *Neural Information Processing Systems*, 2007, pp. 2352-2360, 2010.
[9] Wasserstrom, E. Numerical solutions by the continuation method. *SIAM Review*, 15 (1): 89-119, 1973.
[10] Wilson, A., Fern, A., and Tadepalli, P. A Bayesian approach for policy learning from trajectory preference queries. In *Neural Information Processing Systems*, pp. 1142-1150, 2012.

V. From Turing to the cloud : que peut-on calculer dans un système réparti ?

A. Un constat : un apport fondamental du XXe siècle

Depuis les travaux d'Alan Turing, trois des apports majeurs de l'informatique résident dans la théorie des langages (hiérarchie de Chomsky), la théorie de la calculabilité et la théorie de la complexité. Les très nombreux résultats dans ces domaines ont eu pour conséquence la compréhension profonde de la nature du calcul et sa maîtrise [4]. Celles-ci sont aujourd'hui des constituants incontournables et centraux du monde numérique. Ce sont également ces apports et ces résultats qui ont été les éléments fondateurs de l'informatique en tant que science.

B. Un défi majeur pour le début du XXIe siècle : le calcul réparti

Non seulement le monde est réparti, mais de plus en plus d'applications sont réparties. Il suffit de penser aux bases de données, à la plupart des systèmes embarqués, etc., ainsi qu'aux applications « nuageuses » (*cloud computing*).

Il est donc important, voire crucial, de donner des bases saines au monde du calcul réparti. Une des premières questions qui se pose est la suivante : que peut-on calculer dans un système réparti en présence d'adversaires tels que l'asynchronie et les défaillances ? ou en présence d'adversaires comme la mobilité, la consommation d'énergie, etc., qui sont particulièrement cruciaux dans le domaine des réseaux de robots ou de capteurs ?

Il existe aujourd'hui une multiplicité de modèles de calcul réparti. Cette multiplicité est due non seulement à l'imagination des chercheurs mais surtout à la pléthore d'applications (il suffit de penser aux systèmes répartis embarqués, aux réseaux définis par logiciel – *software-defined networks* –, et aux applications sur les réseaux sans fil). La compréhension profonde des problèmes liés à la nature du calcul réparti nécessite cependant un recul, une réflexion de longue durée que ne permet pas une approche contractuelle de la

recherche et l'urgence qu'elle engendre. Un résultat fondamental et unificateur est toutefois connu à ce jour. Ce résultat est le suivant [2] : une machine de Turing fiable est plus puissante (au sens de la calculabilité de Church-Turing) qu'un réseau de machines de Turing connectées par des canaux point-à-point, asynchrones et fiables, dès lors qu'une machine peut s'arrêter de fonctionner de facon inopinée. Ce résultat est à la base de nombreux travaux actuels sur la calculabilité répartie. Il a donné naissance à des notions d' « hypothèse minimale » (ou de borne inférieure) permettant de résoudre des problèmes répartis particuliers, voire des classes de problèmes (à titre d'exemples, les notions d'oracles répartis tels que les détecteurs de fautes ou l'utilisation de *common coin* fondés sur les nombres aléatoires, entrent dans cette approche).

Dans un tel contexte, un ***défi majeur*** **consiste à établir les fondements du calcul réparti**, comme l'ont fait pour le calcul séquentiel nos insignes prédécesseurs dans les années 1936-1975. De même que les théoriciens de la physique sont à la recherche de la « Théorie de la Grande Unification », il s'agit de mettre de l'ordre et tenter de trouver le « Grand Modèle Unificateur du Calcul Réparti », c'est-à-dire le modèle des applications du futur. Il s'agit là d'un prérequis indispensable si l'on veut être ensuite capable de fournir aux programmeurs les outils conceptuels et logiciels (tels que langages, compilateurs, outils de synchronisation et de distribution, méthodologie, etc.) qui leur permettront de mener à bien la réalisation saine de leurs applications réparties. Le calcul réparti sera alors compris et maîtrisé comme l'est aujourd'hui le calcul séquentiel.

Dans la liste de références ci-dessous, les ouvrages [5, 7, 8] offrent une vision plus large de la thématique exposée ci-dessus. Les publications qui ne sont pas listées dans le texte donnent quelques points d'entrée sur des sujets particuliers.

Références

[1] Fraigniaud P., Korman A., and Peleg D., Toward a complexity theory for local distributed computing. *Journal of the ACM, 60(5), Article 35*, 2013.

[2] Fischer M.J., Lynch N.A., and Paterson M.S., Impossibility of distributed consensus with one faulty process. *Journal of the ACM,* 32(2):374-382, 1985.

[3] Goldreich O., *On our duties as scientists.* http://www.wisdom.weizmann.ac.il/odedg/on-duties.html, 2009.

[4] Harel D. and Feldman Y., *Algorithmics, the spirit of computing*. Springer, 2012.

[5] Herlihy M.P., Kozlov D., and Rajsbaum S., *Distributed computing through combinatorial topology,* Morgan Kaufmann/Elsevier, 2014 (ISBN 9780124045781).

[6] Herlihy M.P., Rajsbaum S., and Raynal M., Power and limits of distributed computing shared memory models. *Theoretical Computer Science,* 509:3-24, 2013.

[7] Lynch N.A., *Distributed algorithms.* Morgan Kaufmann, 1996.

[8] Raynal M., *Concurrent programming: algorithms, principles and foundations.* Springer, 2013 (ISBN 978-3-642-32026-2).

[9] Raynal M., What can be computed in a distributed system? *Proc. Workshop "From Programs to Systems: The Systems Perspective in Computing" in honor of Professor Joseph Sifakis,* Springer LNCS 8415, pp. 209-224, 2014.

[10] Raynal M., From Turing to the clouds. *Proc. 21th Int'l Colloquium on Structural Information and Communication Complexity,* Springer LNCS to appear, Takayama (Japan), July 2014.

VI. Sécurité prouvée

Les questions de sécurité informatique prennent de plus en plus de place, que ce soit dans les systèmes embarqués (cartes à puce, cartes RFID, passeports ou CI électroniques, etc.), les réseaux filaires ou mobiles (téléphonie par exemple), les serveurs, ordinateurs, périphériques, les systèmes de pilotage (dans l'industrie, dans les transports), etc.

La sécurité informatique est citée comme un des défis sociétaux prioritaires dans le programme H2020. Ses champs d'application s'étendent sans cesse. Le *cloud computing* par exemple pose de nouveaux défis dans ce domaine.

Plusieurs niveaux d'attaque sont possibles :
- sur le matériel, par mesure d'intensité, injection de fautes, etc.,
- sur la cryptographie, par cryptanalyse des primitives, comme le chiffrement ou les fonctions de hachage,
- sur les protocoles, soit que les programmes soient incorrects, soit que les spécifications elles-mêmes comportent des failles logiques,
- sur les systèmes d'exploitation, en s'appuyant sur des failles des divers systèmes.

Les attaques matérielles occupent une part importante mais ne sont pas considérées ici car, jusqu'à présent, les travaux de recherche dans ce domaine utilisent des techniques éloignées de celles qui sont employées dans les trois autres domaines.

A. Etat de l'art

Deux types de méthodes sont mises en œuvre pour contrer ces attaques : des méthodes de détection et des méthodes de prévention.

Les méthodes de détection d'attaques (en temps réel) restent limitées car elles ne peuvent détecter que des attaques déjà connues ou proches d'attaques déjà connues en repérant des séquences d'évènements anormales. C'est le domaine de la détection d'intrusion, pour lequel un vrai défi est justement de permettre la détection de nouvelles attaques.

Du côté de la prévention, on peut à nouveau distinguer deux types d'approches. La première, de type *hacker* consiste à chercher des attaques puis à modifier les primitives cryptographiques, les protocoles, les contrôles d'accès pour prévenir les attaques découvertes. Mais cette approche, assez bien valorisée aux États-Unis, relève peu de la recherche scientifique.

Notons aussi que les questions de sécurité présentent des spécificités : par exemple, le test statistique du logiciel est mal adapté aux propriétés de sécurité, car on peut penser qu'un attaquant choisira toujours le pire cas, celui qui n'est pas couvert par le test.

L'autre approche, beaucoup plus satisfaisante, consiste à prouver les primitives cryptographiques, les protocoles, les systèmes d'exploitation. Ces preuves sont plus ou moins formelles mais permettent d'accroître notre confiance.

Dans le domaine des systèmes d'exploitation, par exemple, si jusqu'ici la sécurité reposait beaucoup sur le cloisonnement, des projets de systèmes certifiés ont commencé à être menés à bien : *Trustworthy Systems* en Australie, *CRASH/SAFE* aux États-Unis.

Un des aspects les plus délicats dans les preuves de sécurité est d'identifier clairement les hypothèses. Les preuves sont souvent effectuées dans des modèles dont la pertinence n'est pas évidente, ce qui conduit parfois au paradoxe d'une attaque sur un protocole prouvé... Mais évidemment il s'agit d'une attaque qui ne rentre pas dans le cadre du modèle dans lequel a été effectuée la preuve.

B. Défis

Le défi scientifique est d'accroître significativement notre confiance dans la sécurité informatique des communications, des systèmes d'exploitation, des données, des primitives cryptographiques qui sont utilisées. Ceci passe par l'identification et la formalisation des propriétés de sécurité (ce n'est pas si simple, même dans des domaines formels comme la cryptographie) mais aussi de ce que c'est qu'un attaquant. À quoi a-t-il accès ? Quelles sont ses ressources ? Plusieurs modèles sont utilisés, suivant les communautés, mais les modèles ne sont pas toujours cohérents et pas toujours adaptés à une preuve formelle.

Il faut ensuite ou bien prouver des systèmes, des politiques de sécurité, des primitives cryptographiques existantes, ou bien revoir la façon de les construire ou de les programmer pour faciliter la preuve de leur sécurité. C'est plutôt dans cet esprit que sont

conçus par exemple les projets de systèmes d'exploitation mentionnés plus haut.

Un dernier défi concerne un type de propriété de sécurité qui prend de plus en plus d'importance : le respect de la vie privée. Comment s'assurer que nos activités ne sont pas tracées ? Comment s'assurer que nos données personnelles ne sont pas divulguées, par recoupement de statistiques ? Ces propriétés ont la particularité de s'exprimer comme l'indistinguabilité de deux expériences : l'une menée avec les données réelles et l'autre avec des données modifiées. Prouver ce type de propriété est hors de portée de la plupart des outils de vérification actuels. À nouveau, un défi est de mettre au point des outils formels permettant de prouver le respect de la vie privée dans une grande variété d'applications.

VII. Complexité paramétrée

Certains problèmes de calcul s'expriment naturellement comme des fonctions f qui dépendent d'au moins deux entrées M et φ, de nature différente, et qui ne changent pas aussi fréquemment l'une que l'autre (pour un même M plusieurs φ doivent être pris en compte). Entrent dans cette catégorie des problèmes variés, issus de divers domaines de l'informatique, comme la vérification de modèles, l'évaluation de requêtes dans les bases de données, le diagnostic de systèmes, la configuration de produits combinatoires, la planification d'actions, etc. Par exemple, pour une base de données M (ou un programme M), on s'intéresse aux résultats de f pour plusieurs requêtes φ (ou à vérifier plusieurs propriétés φ du programme), ou à l'énumération des réponses à la requête. En configuration de produits, M est un réseau de contraintes qui représente les produits réalisables et φ traduit les choix de l'utilisateur : il s'agit entre autres choses de pouvoir énumérer efficacement (c'est-à-dire avec un délai polynomial) les produits de coût minimum correspondant aux choix de l'utilisateur.

Or, la théorie de la complexité classique, qui voit chaque entrée comme un tout, ne rend pas compte finement de la complexité du calcul de f. C'est particulièrement le cas quand le calcul de $f(M, φ)$ doit être réalisé pour une même valeur de la première entrée M et de nombreuses valeurs de la seconde φ.

A. État de l'art

Pour répondre à cette difficulté, la théorie de la compilation des connaissances (voir par exemple [1]) et celle de la complexité paramétrée (voir par exemple [2]) ont été développées toutes les deux depuis une quinzaine d'années, quoiqu'indépendamment l'une de l'autre.

En compilation de connaissances, il s'agit d'évaluer dans quelle mesure un pré-traitement C (la compilation) d'informations M peut être utile pour améliorer la résolution d'un problème en termes de temps de calcul (souvent il s'agit d'assurer qu'il pourra être résolu en temps *on-line* polynomial), et de déterminer les pré-traitements C utiles (en particulier, les langages vers lesquels compiler M).

Ces deux questions-clés sont à l'origine des travaux sur le sujet, ancrés d'une part sur la notion de compilabilité et d'autre part sur celle de carte de compilation. L'objectif de garantir des temps de réponse *on-line* constitue un enjeu particulièrement important pour une vaste palette d'applications, de l'interaction avec des utilisateurs pour des applications déployées sur la toile, à la prise de décision pour des systèmes embarqués.

La complexité paramétrée a été introduite pour rendre compte de façon plus fidèle de la complexité de problèmes de calcul, en ne considérant pas exclusivement la taille de l'entrée, mais aussi un ou des paramètres lié(s) à cette entrée.

Il s'agit souvent de paramètres qui expriment une dimension structurelle de l'entrée, comme la largeur d'arbre quand l'entrée est un graphe de contraintes. Par exemple, le problème de vérification de modèles (*model checking*) est **PSPACE-**difficile, mais est linéaire à modèle fixé : il est dans la classe **FPT**, qui est l'analogue du temps polynomial pour la complexité paramétrée. Au contraire, le problème de l'évaluation de requêtes conjonctives, est **W**[1]-difficile, « l'équivalent » de **NP**-difficile pour la complexité paramétrée.

La complexité paramétrée constitue maintenant un des sous-domaines principaux de la théorie des algorithmes. Elle a connu de nombreux développements ces dernières années, notamment en théorie des graphes, et des pistes intéressantes reliant la logique, les problèmes de satisfaction de contraintes et la complexité paramétrée ont été mises au jour.

B. Défi

Néanmoins, il semble pertinent d'étendre et de marier la théorie de la compilation des connaissances et celle de la complexité paramétrée pour leur permettre de mieux rendre compte de la complexité « réelle » des problèmes, lorsqu'un prétraitement d'une partie M de l'entrée est envisageable.

Par exemple, dans la théorie de la compilation des connaissances, il est exigé que $C(M)$ soit toujours de taille polynomiale en celle de M. Cette contrainte rend de nombreux problèmes théoriquement non compilables en temps polynomial. Par exemple, en configuration, le problème de déterminer si les choix utilisateurs φ et l'ensemble des produits réalisables sont compatibles. Or, en pratique, il suffit que la taille de $C(M)$ reste « raisonnable » pour que la compilation puisse se révéler utile. Il serait donc intéressant de revisiter la théorie de la compilation des connaissances et de déterminer comment les résultats obtenus (classes de compilation, réductions) évoluent lorsque la contrainte de polynomialité sur la taille de $C(M)$ est relaxée en contrainte de *polynomialité à paramètre fixé*. Selon le paramètre retenu, la compilabilité du problème peut, en effet, changer. Ainsi, pour l'exemple ci-dessus, il est possible de compiler un graphe de contraintes M en un diagramme de décision $C(M)$ équivalent dont la taille est exponentielle seulement dans la largeur de chemin de M (et pas dans la taille de M lui-même). Etant donné $C(M)$, le problème de la cohérence des choix utilisateurs φ est dans **P**. En pratique, le calcul de tels paramètres permet de décider de tenter ou pas une compilation de M dans tel ou tel langage.

Souvent, ces paramètres sont eux-mêmes difficiles à calculer, mais ils sont parfois approximables ou ils possèdent des algorithmes **FPT**. Dans tous les cas, leur calcul peut être réalisé *off-line* (ils ne dépendent que de M).

Par ailleurs, le choix d'un langage L dans lequel exprimer $C(M)$ s'appuie typiquement sur son expressivité, son efficacité spatiale (c'est-à-dire sa capacité à représenter de l'information en utilisant plus ou moins d'espace) et son efficacité temporelle (les opérations d'intérêt calculables en temps polynomial lorsque l'information est représentée dans L).

Ces propriétés restent à gros grain : pour pouvoir les utiliser au mieux pour choisir un langage L de compilation, il serait intéressant de les préciser en ayant ici aussi recours au cadre de la complexité paramétrée ; en effet, quand une opération n'a pas d'algorithme connu en temps polynomial mais possède un algorithme **FPT**, il est tout à fait envisageable d'utiliser ce dernier en pratique pour tous les M pour lesquels le paramètre considéré reste petit.

Enfin, même si M est fixé, sa taille peut être très importante ; c'est le cas des bases de données ou des modèles en *model checking*.

En pratique, on calcule donc un index ou résumé du paramètre. On peut supposer que ce calcul est effectué une fois pour toutes. L'index ne doit pas être trop volumineux mais il doit être suffisant pour calculer $f(M, \varphi)$ de

manière efficace (par rapport à φ). Le calcul de « bons » index est d'ailleurs au cœur des technologies des moteurs de recherche.

Un des aspects du défi consiste aussi à mettre au point une théorie de la complexité des fonctions de plusieurs paramètres, qui rende compte plus finement de la dépendance des paramètres.

Références

[1] Cadoli M., and Donin F.M., A survey on knowledge compilation. *AI Commun.*, 10(3,4) :137-150, 2013.
[2] Downey R.G., and Fellows M.R., *Parametrized complexity*, Springer, 1999.

VIII. Géométries numériques : représentations, mesure et calcul

Plusieurs domaines scientifiques manipulent des formes par le biais de leur représentation numérique dans un ordinateur :
- la ***modélisation 3D*** (créer des formes),
- l'***acquisition 3D*** (reconstruire des formes à partir de mesures et d'échantillonnage d'un objet réel),
- la ***visualisation scientifique*** (regarder des formes et mettre en évidence certaines de leurs caractéristiques),
- la ***simulation numérique*** (prédire le comportement de la réalisation physique d'une forme sous certaines conditions, le tout modélisé le plus souvent par une équation aux dérivées partielles),
- et la ***fabrication assistée par ordinateur*** (piloter une machine pour construire la forme).

Chacun de ces domaines représente la forme et ses caractéristiques à l'aide d'une représentation formelle jouant le rôle de « langage ». Ce « langage » consiste par exemple en un ensemble de structures de données, de spécifications algébriques, ou de fonctions mathématiques, et comporte des algorithmes et méthodes permettant de manipuler ces représentations formelles.

A. Verrous scientifiques

Chacun des domaines cités (modélisation, visualisation, simulation) s'intéresse à différentes caractéristiques de la forme et se fonde sur la représentation formelle la mieux adaptée aux caractéristiques qui l'intéressent. Par exemple, la visualisation scientifique représente souvent la forme par un volume discrétisé spatialement (voxels, octrees, etc.) qui permet facilement d'extraire des surfaces implicites. La modélisation 3D utilise des surfaces polynomiales et fractions rationnelles (Splines, NURBS, etc.) bien adaptées à la description de pièces mécaniques. La simulation numérique quant à elle utilise des maillages munis de bases de fonctions (éléments finis, volumes de contrôle, différences finies, etc.) pour discrétiser des équations aux dérivées partielles.

Appliquer les algorithmes et méthodes de différents domaines aux mêmes données et à la

même forme se heurte à la diversité et à l'incompatibilité de ces représentations. À première vue, les représentations numériques utilisées ont en commun la caractéristique d'être **discrètes**, à savoir de se réduire à un nombre fini de degrés de libertés, manipulables en machine. Toutefois, ***la nature même de cette discrétisation est fondamentalement différente*** suivant les domaines.

- Discrétisation de l'espace : la forme est représentée par décomposition spatiale, comme un ensemble de cellules interconnectées, régulières (voxels, octrees) ou non (maillages).

- Discrétisation des paramètres : la forme est représentée par une famille de fonctions (souvent des polynômes ou des fractions rationnelles) et la discrétisation correspond aux coefficients d'une combinaison linéaire de ces fonctions.

Nous pensons qu'un ensemble de problèmes scientifiques intéressants se situe précisément à la frontière entre ces deux notions de discrétisation. En particulier, la formalisation des propriétés intéressantes – telles que la topologie, la continuité, le caractère lisse, les grandeurs différentielles comme la courbure – utilise un vocabulaire emprunté au ***continu*** (espaces topologiques, homotopie/homologie, fonctions et formes différentielles). Il manque encore des outils théoriques pour mieux comprendre le lien entre la version discrétisée (des deux manières : espace ou paramètres) et l'origine continue de ces notions. En particulier, comprendre quelles propriétés doivent être préservées par la discrétisation et sous quelles conditions, permettra d'inventer de nouvelles structures de données et algorithmes (reconnaissance de formes, mesures géométriques, optimisation de formes et de maillages, etc.) s'affranchissant des frontières entre les domaines.

B. Impact et positionnement

Les progrès récents dans les techniques d'acquisition (capteurs, scanners 3D) et de fabrication additive (ou « impression 3D ») permettent d'envisager des processus de fabrication industriels jusqu'alors impossibles : les techniques d'optimisation de formes calculent la forme optimale permettant d'optimiser la résistance mécanique d'une pièce. Cette forme optimale présente des structures internes (d'aspect presque « biologique », similaires par exemple aux structures osseuses) impossibles à fabriquer avec les techniques d'usinage classiques mais désormais accessibles par fabrication additive (des imprimantes 3D permettant de fabriquer des pièces en métal sont à présent disponibles). Lever le verrou scientifique de la discrétisation et de la représentation des formes permettrait de rendre les processus d'acquisition, de simulation numérique, d'optimisation de formes et de fabrication parfaitement compatibles. Ceci jouera un rôle central dans les processus de conception et la fabrication des objets de demain, au centre de l'usine du futur et élément-clé de la compétitivité des entreprises.

La modélisation géométrique d'objets physiques est également un thème critique de la robotique à la charnière entre les fonctions perceptives et les fonctions décisionnelles comme la manipulation d'objets ou la planification de mouvement sans collision.

Ce défi se situe clairement à l'interface entre l'INS2I, l'INSMI et l'INSIS.

IX. Optimisation convexe et relaxations semi-définies : vers une technologie pour les sciences de l'information

A. Contexte : une aventure scientifique à l'interface entre automatique et mathématiques

Dans les années 1980, les travaux en optimisation sont appliqués pour résoudre des problèmes d'ingénierie, notamment d'automatique, mais sans réellement exploiter leur structure mathématique. Dans les années 1990 l'aventure a commencé avec le développement d'algorithmes de résolution numérique de problèmes d'optimisation conique convexe sur la théorie des fonctions barrières auto-concordantes. La même période correspond à l'avènement de la commande robuste, une branche de l'automatique permettant de prendre en compte de manière explicite la présence d'incertitudes dans les modèles. Les automaticiens prennent conscience que de nombreux problèmes de commande robuste peuvent se formuler comme des problèmes de programmation linéaire sur le cône des matrices semi-définies c'est-à-dire des problèmes sur les inégalités matricielles linéaires, ou problèmes d'optimisation semi-définie (SDP). Toujours dans les années 1990, en parallèle, en théorie des graphes et en mathématiques discrètes, on observe que des bornes de très bonne qualité sur les optima de problèmes difficiles d'optimisation combinatoire peuvent être obtenus en résolvant des SDP. L'intérêt accru des automaticiens et des théoriciens des graphes pour l'optimisation SDP incite la communauté de programmation mathématique à développer de nouveaux algorithmes et des logiciels. Au début des années 2000, de nombreux solveurs SDP sont disponibles dans le domaine public et plusieurs interfaces pour les ingénieurs sont développées. Bénéficiant des avancées théoriques en commande robuste, la communauté de programmation mathématique applique ces idées et développe la théorie de l'optimisation robuste.

À ce stade de l'aventure certains chercheurs formulent l'idée que les SDP peuvent devenir une « technologie ». Le terme est alors utilisé à la fois pour se féliciter de l'arrivée de solutions logicielles nouvelles, et pour signifier que les démarches développées dans un champ scientifique seraient généralisables à nombre d'autres problèmes. Il s'agit à ce stade plus d'une conjoncture que d'une prédiction.

B. Méthodologie : hiérarchies de relaxations semi-définies

Au tournant du XXIème siècle, les liens entre automatique et optimisation prennent un nouvel essor et ouvrent la voie à la généralisation des méthodes. Il s'agit là essentiellement de travaux qui, grâce à l'existence de solveurs SDP, ont permis d'exploiter certains résultats puissants de géométrie algébrique réelle, résultats issus de retombées du 17ème problème de Hilbert sur la représentation de polynômes positifs comme des sommes de carrés et le problème dual des moments.

Initialement développée dans le cadre de l'optimisation polynômiale de dimension finie, une nouvelle méthodologie a vu le jour pour résoudre de nombreux problèmes non-convexes. Dans un premier temps, il s'agit d'une reformulation comme problème linéaire dans les cônes duaux des fonctions continues et mesures non-négatives. Dans un second temps, ce problème linéaire est résolu numériquement par une hiérarchie de relaxations convexes, à savoir des problèmes linéaires semi-définis de dimension finie mais croissante, avec garantie de convergence.

Actuellement, dans le champ de l'automatique, ces travaux sont étendus à l'évaluation de performance (analyse de robustesse des systèmes non-linéaires incertains, avec retards, saturations, etc.) ainsi qu'à certains problèmes d'identification et de contrôle optimal. La notion de mesure d'occupation permet de résoudre des problèmes de contrôle optimal non-convexes à l'aide de hiérarchies semi-définies. Des travaux visent à combiner les avancées théoriques en analyse convexe, calcul des variations, contrôle des équations aux dérivées partielles, transport optimal avec les avancées en optimisation numérique.

Mais l'intérêt pour les relaxations semi-définies va au-delà du cercle initial de l'automatique et de l'optimisation. Toutes les sciences de l'information sont potentiellement

concernées. Les informaticiens théoriciens se sont aussi intéressés à l'optimisation polynomiale car elle permet de définir des hiérarchies de relaxations semi-définies pour une approximation efficace des problèmes d'optimisation combinatoires difficiles. Un sujet de recherche actif actuellement concerne l'utilisation des hiérarchies semi-définies pour évaluer la complexité des problèmes combinatoires. Des travaux abordent la recherche d'invariants dans les programmes informatiques. De surprenantes connections avec les techniques quantiques en complexité algorithmique sont aussi étudiées activement. Les relaxations semi-définies ont montré leur efficacité en cryptographie pour les codes avec corrections d'erreur. Il est vraisemblable que d'autres problèmes difficiles des sciences de l'information pourront être abordés sous cet angle nouveau.

C. Vers une « technologie »

L'ensemble de ces travaux utilisant les relaxations semi-définies illustre que la démarche est plus qu'un résultat isolé. Il est permis de penser qu'elle puisse évoluer vers une « technologie » au double sens : (1) d'une méthodologie applicable à un cadre relativement général, et (2) d'une approche fournissant des outils logiciels de résolution effective.

X. Le code numérique du vivant

L'ADN est le code numérique du vivant. Il s'écrit sur un alphabet de quatre lettres ou nucléotides (A, C, G, T). L'ADN a une structure en double hélice, contenant deux séquences appariées, l'une se déduisant de l'autre. Les génomes des espèces sont constitués de longues molécules d'ADN, contenant des milliers, des millions, voire des milliards de paires de bases (ou paires de nucléotides appariés). La taille et le contenu informationnel des génomes varient considérablement. Un virus comme le VIH contient 10000 paires de bases, tandis que le génome humain en contient trois milliards. Ces chiffres peuvent paraître élevés, notamment lorsqu'il s'agit d'acquérir ces données (plus de 15 ans pour le premier génome humain). Ils sont en réalité très faibles dès lors qu'on raisonne en termes de complexité. Il est par exemple étonnant qu'on ne comprenne pas mieux le fonctionnement du VIH, alors que son code tient dans moins de 5 pages comme celle-ci. Le génome humain tient sur un CD, alors qu'il code toute notre complexité, notamment celle de notre cerveau.

On fait souvent l'analogie entre ADN et programme informatique. L'ADN est le logiciel du vivant, la cellule en est le hardware. Cette analogie a ses limites, nous le verrons, mais elle résiste bien aux expériences et au temps. On a récemment remplacé l'ADN d'une bactérie par un ADN synthétique issu des bases de données publiques, et la bactérie ainsi reconstituée semblait parfaitement fonctionnelle. On est capable de « reprogrammer » les génomes de certaines espèces pour leur faire synthétiser des molécules d'intérêt, médicaments ou autres. Les virus comme le VIH ajoutent des instructions à notre propre programme (génome) pour faire travailler les cellules infectées à leur profit.

Cette analogie a cependant ses limites, lesquelles donnent matière à penser pour les chercheurs en informatique et sciences de l'information. Nous avons déjà parlé du caractère compact des génomes. Le code de ceux-ci est aussi remarquablement flexible : en changeant un caractère, on change très rarement la fonction et le phénotype, au point que des génomes ou des gènes très différents

en termes de séquence peuvent correspondre à des objets biologiques quasi-identiques. Cette flexibilité autorise l'évolution du vivant : le code se modifie et se diversifie progressivement au cours du temps et des reproductions successives, ceci de manière graduelle et essentiellement aléatoire, les génomes les mieux adaptés étant retenus par la sélection naturelle. Cependant l'ADN est une molécule remarquablement stable d'un point de vue biochimique. Ainsi, on peut encore à ce jour lire le génome de l'homme de Neandertal, qui date d'environ 30 000 ans. Un autre aspect fascinant est que le code lui-même (l'ADN) est à l'origine du hardware (les protéines et la machinerie cellulaire), lequel assure la survie, l'évolution et la reproduction du code. Finalement, il faut garder à l'esprit que la cellule elle-même contient une information importante dite « épigénétique », encore très mal cernée et dont l'étude fait partie des sujets centraux de la biologie d'aujourd'hui.

Les révolutions technologiques de ces dernières années ont permis d'élaborer des techniques de séquençage (ou lecture) des génomes à coût très faible. Quelques semaines et mille dollars suffisent pour un génome humain, là où au tournant des années 2000 il avait fallu 15 ans et des milliards de dollars pour séquencer le premier génome de l'homme. En conséquence, et en raison de leur contenu informationel très riche, le volume des données de séquences croît exponentiellement. À celles-ci s'ajoutent des données sur la structure spatiale des molécules, sur leur capacité à interagir, sur leur présence et leur abondance au sein de la cellule dans des conditions données, et bien d'autres encore. Ces données « post-génomiques » sont d'une grande importance pour donner leur sens aux données génomiques.

Cette masse de données complexes pose des défis majeurs à l'informatique et aux sciences de l'information. Les disciplines concernées sont la bioinformatique, la biologie computationnelle et la biologie des systèmes, celles-ci faisant appel à de nombreuses spécialités fondamentales comme l'algorithmique, la modélisation, l'automatique et les bases de données. Ces spécialités identifient d'ailleurs clairement le traitement et l'analyse des données biologiques parmi leurs objectifs. Trois directions de recherche peuvent être distinguées. Nous commencerons par celles qui ne sont pas les plus visibles aujourd'hui, mais qui mériteraient de l'être ou de le redevenir.

(1) En premier lieu, comprendre l'ADN en tant que système de codage, avec ses spécificités soulignées plus haut, pose des questions fondamentales à l'informatique théorique et aux théories de l'information. Comment concilier cette compacité et cette flexibilité avec l'immense complexité du vivant ? Les premiers travaux dans cette direction, il y a une quinzaine d'années, appuyés sur la théorie de la complexité, n'ont pas été concluants. Les progrès de la théorie de l'information et du calcul comme ceux de la biologie, et la disponibilité de données considérablement plus nombreuses sont un encouragement à reprendre ces questions, qui demeurent fascinantes et dont la compréhension est la clé des applications les plus novatrices.

(2) Ensuite, il faut poursuivre et approfondir l'exploitation des propriétés de l'ADN et son fonctionnement dans des approches bio-inspirées. Les algorithmes génétiques optimisent des fonctions complexes en s'inspirant des mécanismes de mutation, recombinaison et sélection du vivant. Divers modèles et méthodes de calcul hautement parallèles ont été proposés sur la base des propriétés d'appariement de l'ADN ; on parle alors de DNA computing. Tout récemment, la faisabilité du stockage de données numériques grâce à l'ADN a été démontrée. Un texte est retranscrit dans l'alphabet A, T, G et C, avec l'adjonction éventuelle de codes correcteurs d'erreur, puis l'ADN correspondant est synthétisé et stocké. Les coûts et temps d'écriture et de lecture sont bien sûr plus importants qu'avec un disque dur, mais l'information devrait demeurer pour des milliers d'années, si l'on en croît l'exemple de l'homme de Neandertal.

(3) Aujourd'hui le problème majeur, qui mobilise des forces sans cesse croissantes en informatique, est de traiter et analyser les données de séquence. Il s'agit de comprendre comment l'information des séquences s'exprime, et comment leurs produits (protéines et ARN) se structurent et interagissent entre eux pour faire fonctionner et évoluer la cellule. La dimension évolutive joue un rôle clef. Comprendre les objets biologiques, c'est comprendre comment ils ont évolué et se sont constitués et spécialisés. Les approches comparatives permettent de révéler les parties conservées au cours de l'évolution,

et donc fonctionnellement importantes, et celles pouvant être liées à une adaptation récente. Ainsi, pour comprendre le génome humain on a séquencé les génomes proches, comme celui du chimpanzé, avec en corollaire des problèmes informatiques ardus pour comparer ces (grands) génomes. Les données de séquences abondent dans de très nombreux domaines, leurs analyses sont exploitées pour des applications très diverses, de biologie fondamentale bien sûr, mais aussi en médecine ou en environnement, et dans de nombreux secteurs de la vie quotidienne.

Nous dressons maintenant un panorama rapide des différentes recherches en informatique et sciences de l'information mises en œuvre pour dépasser les obstacles qui se présentent.

- La première difficulté est le passage à l'échelle. Les volumes de données croissent plus vite que la puissance de calcul des ordinateurs (jusqu'à présent gouvernée par la fameuse loi de Moore). Pour stocker les données et les exploiter, des techniques toujours plus sophistiquées d'indexation sont actuellement développées, dans la continuité des arbres des suffixes ou, par exemple, à partir de techniques probabilistes de hachage en grande dimension.
- Pour analyser ces données, l'algorithmique, souvent discrète, joue un rôle central : algorithmique du texte en premier lieu, pour comparer des gènes ou des génomes, ou découvrir des motifs ou signaux le long des séquences ; puis algorithmique des graphes et des réseaux, qui sont utilisés dans de nombreuses représentations : interactions entre objets biologiques, voies métaboliques, réseaux de régulation génique, contacts et appariements au sein des molécules biologiques, etc.
- La modélisation joue un rôle essentiel dans la compréhension des données biologiques, en permettant de tester des hypothèses et de proposer des explications de plus en plus complètes des observations expérimentales. Les modèles sont souvent statistiques ; les modèles de Markov et Markov cachés ou leurs généralisations sont fréquemment utilisés et nécessitent la mise en place de procédures efficaces d'apprentissage. Les modèles sont aussi spatiaux et géométriques, avec les algorithmes afférants, dès lors qu'on s'intéresse à la structure des molécules biologiques, qui est étroitement liée à leur fonction.
- Finalement, l'ensemble des données et des analyses impose de travailler sur les bases de données et de connaissances, ainsi que sur l'intégration, dans ces bases, de données fortement hétérogènes, de qualité très variable, avec des sources nombreuses et réparties à la surface du globe. Un objectif essentiel est de mettre en place des workflows scientifiques qui soient reproductibles, réutilisables et capables de tracer la provenance des données qu'ils produisent.

L'importance des enjeux a été bien comprise dans l'appel H2020 avec le programme *Advancing bioinformatics to meet biomedical and clinical needs* dont l'objectif est de faire entrer véritablement les données génomiques et post-génomiques dans le secteur médical. On trouve des programmes équivalents du côté de l'environnement et de l'agronomie. L'informatique est au premier plan, en interface évidente avec les sciences de la vie, mais aussi la physique, les mathématiques et les sciences de l'ingénieur.

XI. Traitement quantique de l'information

L'idée de l'informatique quantique remonte aux années 1980 quand Feynman suggérait qu'une machine exploitant les propriétés quantiques de la matière serait capable de simuler efficacement l'évolution de systèmes quantiques, un problème pour lequel on ne dispose d'aucun algorithme classique efficace. Le champ d'application de l'ordinateur quantique s'est considérablement élargi quand Shor a montré qu'un tel ordinateur pouvait factoriser les entiers et calculer le log discret en temps polynomial, et par conséquent, casser la plupart des crypto-systèmes utilisés aujourd'hui.

En parallèle de l'informatique quantique, s'est également développée la cryptographie quantique dont la sécurité des protocoles ne repose plus sur des hypothèses mathématiques, mais sur les lois de la physique quantique. En particulier, la distribution quantique de clés proposée par Bennett et Brassard en 1984, est une technologie maîtrisée et des crypto-systèmes quantiques sont disponibles dans le commerce.

Plus généralement, après la révolution technologique du milieu du XXe siècle qui a reposé sur le silicium et les lasers, nous sommes en train d'assister à une nouvelle révolution technologique : celle du traitement quantique de l'information, où les lois de la physique quantique ne sont pas perçues comme un obstacle dans la course à la miniaturisation, mais comme une ressource nouvelle offrant des perspectives inégalées en termes de puissance de calcul ou de sécurisation des communications.

A. Défis et verrous scientifiques

La mise au point d'un ordinateur quantique universel se révèle être une tâche extrêmement délicate, la difficulté étant d'isoler les bits quantiques suffisamment bien pour éviter qu'ils ne perdent leur propriétés quantiques, tout en les manipulant pour être en mesure d'effectuer le calcul souhaité. Outre ce problème technologique, les questions qui se posent actuellement sont de savoir d'une part ce qu'on est déjà en mesure de faire avec la technologie disponible, et d'autre part, de préparer l'avenir en comprenant mieux les possibilités offertes par le calcul quantique.

La **cryptographie quantique** est certainement la technologie quantique la plus au point. En effet, il est « relativement » simple d'échanger des ressources quantiques à travers les réseaux de télécommunication usuels (fibre optique, satellite) et de créer ainsi des réseaux de communication quantiques permettant de résoudre divers problèmes cryptographiques importants : la sécurisation des communications, le partage de secret, le tirage à pile ou face à distance. Le principal problème à résoudre est d'augmenter la taille de ces réseaux aujourd'hui limitée à la centaine de kilomètres à cause des pertes en ligne. Très récemment, il a été observé que l'obtention de certaines corrélations quantiques garantit qu'elles résultent de la mesure d'un système quantique bien particulier : cette propriété dite de *self-testing* est à l'origine d'un nouveau paradigme cryptographique, celui des protocoles indépendants de l'appareil de mesure, c'est-à-dire des protocoles résistants à toute forme d'attaque par canal auxiliaire. Mieux comprendre ces protocoles est certainement l'une des questions les plus brûlantes du domaine.

L'**algorithmique quantique** vise à trouver des algorithmes quantiques efficaces. Parmi les questions importantes, on peut citer la généralisation de l'algorithme de Shor pour identifier des sous-groupes cachés, le problème de la simulation efficace de l'évolution temporelle de systèmes quantiques ou la recherche de bons codes correcteurs quantiques avec l'objectif de pouvoir faire du calcul quantique tolérant aux fautes avec un *overhead* raisonnable.

La **théorie de la complexité** vise à mieux comprendre les capacités d'un ordinateur quantique et par exemple à établir si ce modèle de calcul permet de résoudre en temps polynomial des problèmes qui sont NP-difficiles dans le modèle des machines de Turing.

Une question particulièrement pressante est de mieux comprendre la complexité des Hamiltoniens locaux (qui décrivent de nombreux systèmes physiques particulièrement pertinents en pratique, par exemple les supraconducteurs à haute température). En particulier, un effort de recherche notable est aujourd'hui dédié à prouver (ou non) une version quantique du

théorème PCP, dont la version classique est certainement le résultat le plus important de ces 20 dernières années en théorie de la complexité.

B. Impact et positionnement

Les impacts sociétaux sont à court terme en ce qui concerne les aspects cryptographiques, et plutôt à moyen terme pour ce qui est du calcul quantique. La contruction d'un véritable ordinateur quantique, permettant par exemple la mise en œuvre de l'algorithme de factorisation de Schor, constitue aujourd'hui encore un véritable défi technologique. Cependant il faut noter que des « simulateurs quantiques » seront très certainement disponibles beaucoup plus rapidement, et permettront de résoudre de nombreuses questions très pertinentes pour l'industrie (par exemple pour la synthèse de molécules dans l'industrie pharmaceutique).

Les domaines d'applications sont nombreux et peuvent être définis comme l'ensemble des domaines où l'informatique classique est employée.

Ces défis se situent clairement à l'interface entre l'INS2I et l'INP. Néanmoins les champs d'application font de ce défi un problème central de nombreux autres instituts (Biologie, Mathématiques, etc.)

CONCLUSION

L'élaboration de ce rapport a été en partie nourrie par des tables rondes qui ont été organisées régulièrement par le Conseil Scientifique d'Institut et qui ont porté sur les thèmes suivants :

- Intelligence ambiante
- Robotique
- Masses de données
- Systèmes intégrés matériels-logiciels
- Rôle et place des ITA dans les équipes de recherche
- Architecture, systèmes, réseaux
- Conception de logiciels
- Complex networks
- Automatique
- Traitement du signal et des images
- Image, vision
- Intelligence artificielle
- Sécurité informatique

Nous adressons nos remerciements chaleureux à Danuta Dufrat-Chabrière (SGCN), à Dimitri Peaucelle membre du Conseil Scientifique de l'INSIS et invité permanent de notre Conseil, aux participants à ces tables rondes, aux co-auteurs des textes qui ont fourni une première source pour certaines sections de notre rapport, ainsi qu'aux présidents de la section 07 et de la CID 44 puis aux présidentes des sections 06 et 07 qui ont très régulièrement assisté à nos réunions, et aux membres du Conseil qui nous ont quittés en cours de route pour prendre d'autres fonctions : S. Abiteboul, M. Basseville, F. Bassino, Ch. Bessière, Ph. Bidaut, E. Burdet, O. Cappé, E. Colin de Verdière, C. Consel, J.-M. Coron, V. Cortier, J. Coutaz, J.L. Crowley, M. Daumas, M. Debbah, F. Devernay, M. Devy, L. Duflot, B. Durand, J. Fadili, E. Fleury, P. Fraigniaud, P. Garda, A. Girard, D. Gross-Amblard, N. Halbwachs, D. Henrion, J.-P. Jessel, B. Kégl, J.-O. Klein, J.-B. Lasserre, A. Laurent, E. Lazega, B. Lévy, P. Marquis, M. de Mathelin, J.-P. Merlet, D. Monniaux, O. Papini, D. Peaucelle, D. Pointcheval, M. Pouzet, I. Queinnec, P. Prinetto, A. Richard, J.-P. Richard, M. Robert, M.-Ch. Rousset, G. Sassatelli, P. Senellart, D. Simplot-Ruyl, G. Thomas, E. Viennet, P. Zweigenbaum.