Une très vielle science, l'informatique

Maurice Nivat

▶ **To cite this version:**

Maurice Nivat. Une très vielle science, l'informatique. Sciences et technologies de l'information et de la communication en milieu éducatif : Analyse de pratiques et enjeux didactiques., Oct 2011, Patras, Grèce. pp.33-47. edutice-00676195

HAL Id: edutice-00676195
https://edutice.hal.science/edutice-00676195

Submitted on 3 Mar 2012

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

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

Une très vielle science, l'informatique

Maurice Nivat
mnivat@gmail.com

Je crois difficile de comprendre pourquoi l'informatique a pris une telle place dans l'ensemble des activités humaines en apparemment quelque chose comme un demi-siècle et, par suite, de prévoir ce que vont être ses développement futurs si l'on en a pas une vision globale et si l'on ne voit pas que son état actuel est le fruit de siècles voire de millénaire de gestation.

Définir l'information n'est pas très facile. Je propose qu'on la définisse comme tout le savoir, toutes les connaissances, nécessaires ou utiles pour prendre une décision à un instant donné.

Chaque matin, chacun d'entre nous, quand il se réveille prend plusieurs décisions : se lever ou non, s'habiller ou non, déjeuner ou non et il est facile de voir que cette décision nécessite des informations assez nombreuses et de natures très variées. Il faut d'abord répondre à toutes les questions suivantes :

- Où suis-je, dans mon lit chez moi, à l'hôtel en vacances, à l'hôpital ?
- Quelle heure est-il ? Est-ce samedi ou dimanche ?
- Ai-je quelque chose à faire aujourd'hui ? Bientôt ? Dans la demi-heure qui suit ?
- Quel temps fait-il ?
- Suis-je malade ?

Mentalement je me compose un programme d'actions à exécuter dans les moments à venir qui dépend de toutes ces informations : par exemple, sauter de mon lit, me débarbouiller vite fait et m'habiller très rapidement, avaler un café et courir à l'arrêt du bus, tout ça parce que j'ai un rendez-vous professionnel dans une heure et tout juste le temps d'y arriver à temps. Mais cela peut-être aussi : tant pis pour mon rendez-vous, je n'ai pas le temps d'y arriver à l'heure, ou bien il n'est pas si important que ça, je n'ai pas de raison de me presser. Se faire un programme c'est choisir une série d'actions à exécuter dans un certain ordre, ordre qui est entièrement déterminé ou bien partiellement déterminé par des contraintes physiques ou autres que je me dois également de connaître. Je peux choisir de boire mon café avant de m'habiller si je suis chez moi mais je ne peux pas courir à l'arrêt du bus avant de m'être habillé.

Baron, G.-L., Bruillard, E., & Komis, V. (2011). Sciences et technologies de l'information et de la communication (STIC) en milieu éducatif : Analyse de pratiques et enjeux didactiques, Actes du quatrième colloque international DIDAPRO 4 - Dida&STIC, 24-26 octobre 2011, Université de Patras. Athènes : New Technologies Editions, 2011. ISBN 978-960-6759-75-8

Les exemples les plus triviaux montrent que les programmes les mieux établis et respectés peuvent ne pas produire les effets escomptés ; si j'arrive à l'arrêt du bus pour constater qu'il n'y a pas de bus parce que les chauffeurs se sont mis en grève à la suite de l'agression de l'un d'entre eux, je ne serai pas à mon rendez-vous, bien que j'ai bien planifié mes mouvements afin d'y être à l'heure. Peut-être que si j'avais écouté la radio hier soir j'aurais appris cette grève aux « infos », et aurais-je fait le nécessaire pour me réveiller et partir plus tôt en voiture, sauf qu'évidemment le temps qu'il me faut pour aller en voiture au boulot est très variable et risque fort d'augmenter considérablement en cas de grève des transports en commun.

L'informatique (mot forgé par Philippe Dreyfus en 1962) se définit alors comme la science et l'ensemble des techniques qui s'occupent de l'information sous toutes ses formes, de sa collecte, de son stockage et de son utilisation pour établir des programmes d'action. Un programme d'action est une suite d'actions qui peuvent être des actions exécutés les unes après les autres par une machine ou par un être humain ou encore, c'est de plus en plus souvent le cas, exécutées simultanément ou en séquence par un ensemble de machines et d'opérateurs humains tel qu'une équipe, un atelier, une entreprise.

Comme la plupart des sciences, la science informatique s'est construite sur un savoir accumulé par les hommes au cours des âges, savoir empirique, fruit d'expérimentions et d'observations et de tentatives plus ou moins maladroites d'apporter des solutions à des problèmes techniques qui se posaient à eux.

Si l'on fait commencer l'astronomie avec l'installation du premier gnomon permettant de repérer la position du soleil, on doit faire commencer l'informatique avec les premiers algorithmes mis en œuvre qui ont nécessité une connaissance des données à transformer, et des opérations à effectuer, pour réaliser cette transformation. Un très bel exemple nous est fourni par la fabrication d'outils tranchants par nos ancêtres du néolithique, disons entre 50000 ans et 5000 ans avant Jésus Christ.

Ces outils tranchants étaient obtenus en débitant des blocs de silex par percussion : au début du néolithique une percussion frontale perpendiculaire à la surface du bloc à débiter provoquait la division du bloc en deux parties dont un bord était tranchant et donc bien commode pour divers usages que l'on imagine. Avec un bloc d'un kilo de silex on obtenait quelques centimètres de tranchant, à la fin de la période on arrivait à tirer cent mètres de tranchant du même bloc de un kilo ! La courbe de l'évolution de la longueur de tranchant que l'on savait tirer d'un kilo de silex, reproduite dans la réédition récente de l'ouvrage du grand paléontologue que fut

André Leroi-Gourhan, « le geste et la parole » est une superbe exponentielle qui s'arrête brutalement avec l'apparition du métal, bronze d'abord, fer ensuite.

Tous les ingrédients d'un problème informatique sont là : pour arriver à leur fin les fabricants de lames de silex devaient avoir une connaissance approfondie du matériau qu'ils travaillaient et du bloc qu'ils avaient à la main, ils devaient savoir orienter le dit bloc pour modifier l'angle d'attaque, c'est par des coups tangentiels qu'ils parvenaient à détacher des éclats tranchants de plus en plus longs, de plus en plus fins et de plus en plus coupants. Une phase de préparation consistait en la transformation du bloc brut en quelque chose comme un biface et ensuite la place et l'orientation du coup suivant était déterminée par la forme du bloc après détachement du dernier fragment obtenu. Il n'est pas inutile d'aller voir au Grand Pressigny en Touraine le musée consacré à cette activité sur le lieu même d'ateliers qui savaient produire des lames de silex de trente centimètres de long, coupantes comme des lames du meilleur acier de Solingen qui étaient exportées dans toute l'Europe, on en a retrouvé à trois mille kilomètres de là. Au Grand Pressigny on ne peut qu'être vivement impressionné par un savoir-faire évidemment technique plus que scientifique mais qui suppose la maîtrise d'une information précise et d'assez longues suites d'opérations élémentaires à effectuer pour obtenir un certain résultat. Infiniment probablement, nos lointains ancêtres savaient aussi transmettre et les informations nécessaires sur le silex et la forme que pouvaient revêtir les blocs qui leur servaient de matériau et le « programme » décrivant les suites d'opérations, ne serait-ce qu'à des fins d'éducation de jeunes apprentis.

Ce dont je voudrais vous convaincre aujourd'hui c'est que l'informatique, science de l'information, est l'héritière directe de millénaires de travail des hommes pour mettre au point des techniques de collecte, de conservation, de transformation et d'utilisation de l'information. Si l'on peut dater du début des années soixante la naissance de l'informatique moderne et son baptême (le mot informatique a été inventé en 1964) elle a été précédée d'innombrables travaux et recherches dont elle incorpore, explique, rationalise les résultats exactement comme la physique des matériaux incorpore, explique, rationalise des millénaires d'observations, et d'expériences au sujet des matériaux faites depuis que l'homme en utilise ou en fabrique.

La gestation de la science informatique a été plus lente que celle d'autres sciences probablement parce que l'information est quelque chose que l'on a plus de mal à saisir et que les gens, pourtant très nombreux qui en manipulaient, n'avaient pas conscience des liens étroits entre ce qu'ils faisaient dans des domaines différents,

voire des champs culturels complètement disjoints. Le lexicographe, le fabriquant de métier à tisser, le médecin, le bâtisseur de fortifications, le banquier ne se rendaient pas compte que chacun à sa manière collectait et utilisait de l'information car les formes de l'information qu'ils utilisaient étaient extrêmement différentes. Les choses ont changé quand les progrès de la technique ont permis d'unifier les représentations de l'information, d'établir des résultats généraux et des méthodes valables pour toute l'information, indépendamment de ses représentations.

Nous allons examiner successivement les trois grands chapitres de l'informatique telle que je la vois.

1. L'information et ses représentations

L'information est aussi nécessaire à la vie que l'oxygène de l'air que nous respirons et que l'eau dont nous ne pouvons-nous passer plus de quelques heures. Toutes nos actions sont conditionnées par des informations que nous détenons, stockés dans notre cerveau, ou perçues dans l'instant de l'action par nos sens.

La première représentation de l'information est le langage parlé qui est une suite de signaux sonores grâce auxquels nous communiquons entre nous et en particulier échangeons de l'information. Ecrit le langage devient une suite de signes optiques qui permet aussi de communiquer et de véhiculer de l'information. C'est grâce au langage parlé ou écrit que, jusqu'à un passé très récent, circulait ou était stocké l'essentiel de l'information**.**

Mais il est beaucoup d'informations qui ne sont pas ou mal transmises ou par le langage vernaculaire et nous pouvons en citer des exemples.

Une information dont nous nous servons constamment est celle qui concerne notre position à un moment donné, dans un lieu géographiquement défini, dans un cadre géométriquement défini, dans un environnement qui peut faire obstacle à nos mouvements. La vision que nous en avons n'est pas forcément suffisante pour l'usage que nous voulons en faire. Il faut bien pourtant que nous sachions répondre à la question permanente « où suis-je ? » qui détermine pour une large part notre attitude ou notre comportement. Contrairement aux horaires de chemin de fer que beaucoup d'entre nous utilisons aussi, il s'agit là d'une information qui, bien qu'essentielle est assez difficile à décrire et à représenter, donc à transmettre et à exploiter, ainsi qu'en témoignent les innombrables travaux qui lui ont été consacrés récemment, en particulier pour faire se mouvoir un robot équipé de caméras dans un environnement donné. Ce très vieux problème a suscité d'innombrables travaux à travers les siècles : une réponse est la cartographie et je ne sais rien de plus émouvant et fascinant que le travail des cartographes des quatorzième et quinzième

siècle cherchant à représenter l'Afrique et l'Amérique à partir des informations très fragmentaires, souvent fausses et contradictoires émanant des marins, voyageurs et explorateurs qui mettaient les pieds dans ces territoires inconnus.

Un autre problème qui intéresse tout le monde est celui du rangement : il ne suffit pas d'accumuler de l'information, il faut encore pouvoir la retrouver quand on en a besoin ce qui introduit le problème du rangement. Où mettre tel ou tel document dans une administration, ou ranger tel ou tel outil dans un atelier pour être sûr de pouvoir le retrouver si besoin est ; et le retrouver si possible rapidement. Bien avant l'imprimerie existaient déjà des bibliothèques qui conservaient des manuscrits et la question de leur indexation ou catalogage se posait, en fait dans les mêmes termes qu'aujourd'hui. Ce sont des bibliothécaires de l'antiquité qui ont commencé à utiliser les structures arborescentes qui sont encore utilisées de nos jours pour hiérarchiser les fichiers sur nos ordinateurs. Les dictionnaires ont une très longue histoire, bien qu'ils ne remontent guère qu'au Moyen-Âge qui a vu apparaître des glossaires, les premiers dictionnaires de langue, soit du français soit latin français datant de la fin du quinzième siècle (Jean Nicot, Robert Etienne) : du dictionnaire de langue à l'encyclopédie, dictionnaire de faits ou de concepts il n'y a qu'un pas qui fut vite franchi et l'établissement de dictionnaires et encyclopédies a été et est toujours une activité essentielle pour l'organisation, la structuration et la rationalisation de l'information.

Le diagnostic médical, aussi vieux que la médecine, repose sur des informations recueillies par le médecin qui sont de nature très diverses, dires du patient, son aspect ou l'aspect de telle ou telle partie de son corps et de ses secrétions, bruits du cœur ou des poumons et des inventions anciennes comme le stéthoscope de Laennec (datant de 1816 et perfectionné évidemment depuis mais pas tant que ça) sont encore en usage de nos jours.

Les problèmes soulevés par l'information, depuis toujours sont multiples : il y a ceux évidents de collecte, de stockage, de mise à jour mais il y en a beaucoup d'autres dont s'est préoccupée l'humanité : celui du secret et donc du cryptage dans la transmission de l'information, celle-ci pouvant valoir l'or. Celui de la validité, de la cohérence et de la pertinence n'est pas moins important et l'est toujours si l'on estime qu'un pourcentage non négligeable des informations stockées dans nos modernes bases de données sont fausses. Sur un champ de bataille la possibilité pour un combattant, un avion ou une batterie de canons de distinguer entre un ami et un ennemi est essentiel : le passé le plus récent montre que les techniques de camouflage ou d'intoxication par propagation d'informations volontairement

erronées sont assez puissantes pour qu'un camp inflige des dommages à ses propres troupes.

L'omniprésence de l'information et les formes extrêmement diverses qu'elle peut revêtir font que de nombreuses questions qu'elle soulève à tous les niveaux de sa collecte ou de son utilisation ont été au cours du temps résolues aussi de façons très diverses selon les domaines concernés. Il s'en suit que la prise de conscience du fait que l'information en tant que telle puisse être un objet d'étude et d'une science est récente.

2. Les algorithmes et les programmes

Les algorithmes décrivent les suites d'opérations à effectuer pour obtenir un certain résultat, suites d'opérations dépendant des données initiales, c'est-à-dire de l'objet, de la situation ou de l'information à transformer. Nous utilisons tous des algorithmes pratiquement à tout instant de notre vie consciente. Dans notre jargon d'informaticien, les programmes sont les textes qui décrivent des algorithmes et rien n'interdit la programmation dans une langue vernaculaire : tout manuel, tout mode d'emploi, toute recette sont des programmes, c'est dire si les programmes sont nombreux et que l'on n'a pas attendu l'avènement des machines à calculer pour en rédiger, en diffuser et en utiliser. Les programmes peuvent aller de quelques lignes à un ensemble d'épais volumes, tel les programmes d'entretien des avions de ligne.

Comme exemple d'algorithme que tout le monde utilise je prendrai l'algorithme de l'habillage qui va me permettre de faire plusieurs remarques. La première est qu'il s'apprend, on l'apprend aux petits enfants, les bébés ne savent pas s'habiller. La seconde c'est qu'il est très dépendant d'informations comme le temps qu'il fait, l'occupation à laquelle on compte se livrer une fois habillé ou les gens que l'on est susceptible de rencontrer. Plus fondamental est le fait que l'on peut considérer un même algorithme à plusieurs niveaux : pour enfiler un t-shirt il existe des algorithmes que les enfants apprennent et une fois qu'ils en ont appris un l'opération « enfiler un t-shirt » devient une opération élémentaire qu'il n'est plus nécessaire de décomposer en mouvements des bras et de la tête, on peut dire à l'enfant « enfile ton t-shirt » quand il sait le faire. L'algorithme que l'on met en œuvre tous les matins peut être dès lors considéré comme fort simple. Sauf évidemment si l'on a le bras cassé ou l'on souffre de violentes douleurs dans l'épaule. Mais il n'est simple aussi que si le choix des vêtements à mettre est simple et si l'on a l'habitude des dits vêtements. « Je ne sais pas quoi me mettre » est une phrase que l'on entend souvent en aux moins deux circonstances, si le

temps est tout à fait incertain et si la personne doit aller à une réception où elle ne sait pas trop ce qui va se passer ni qui va être là. L'aspect modulaire des algorithmes et des programmes qui sont composés d'opérations réputées élémentaires mais qui peuvent souvent ne l'être pas du tout et être des opérations fort complexes que réalisent d'autres algorithmes dont on ne se soucie pas est tout à fait essentiel.

Toutes nos actions sont régies par des algorithmes, la plupart heureusement appris depuis si longtemps et utilisés si souvent que l'on n'y pense plus, on s'habille, on va prendre un bus ou un métro, on conduit sa voiture sans vraiment réfléchir. L'histoire des algorithmes et des programmes est aussi ancienne que celle de l'information et de ses représentations, les fabricants de lames de silex utilisaient des algorithmes assez sophistiqués si bien que les archéologues ont bien du mal à les reconstituer et l'on débat toujours de la façon qu'avaient les dresseurs de menhirs de mettre debout des pierres qui peuvent peser jusqu'à cent tonnes, évidemment sans aucun appareil de levage.

Cette histoire est complètement inséparable de l'histoire de l'outil qui n'a été au début qu'un prolongement de la main de l'homme avant d'être mis en branle (mais ce, quand même, depuis la plus haute antiquité) par une énergie non humaine fournie soit par l'eau courante soit pas des animaux, bœufs, chiens, chevaux, ânes. L'outil s'insère dans la chaîne des opérations à effectuer en se chargeant de quelques-unes qu'il rend possibles, alors qu'elles ne l'étaient pas, ou plus faciles et chaque outil fait naître une nouvelle algorithmique en modifiant l'ensemble des opérations élémentaires dont on dispose. Si l'on veut bien cesser de penser qu'un algorithme est un objet forcément mathématique ou « numérique », on s'aperçoit sans peine qu'une grande partie de l'activité humaine depuis les origines a consisté à inventer de nouveaux algorithmes ou améliorer des algorithmes anciens pour fabriquer ou obtenir des objets ou des effets plus rapidement, à un moindre coût ou avec une plus grande précision. De l'outil simple on est passé à la machine réalisant des opérations de plus en plus complexes « automatiquement », c'est-à-dire sans intervention humaine et l'on est arrivé au robot qui fait tout tout seul. La tendance millénaire à minimiser la part confiée à l'opérateur humain et à développer la part confiée à l'outil ou la machine dans une chaîne d'opérations aboutit au robot trayeur de vaches que j'ai vu l'autre jour dans le village où je possède une maison de campagne : la vache vient toute seule se mettre en position debout dans un espace approprié attirée par le besoin qu'elle ressent d'être traite quand ses mamelles sont pleines. Là le robot trouve le pis, le lave, applique sur chaque pis un suceur et traie la vache tout en analysant son lait, celui que donne

une vache atteinte de mammite est mis à part. L'éleveur n'a plus rien à faire que surveiller du coin de l'œil et celui qui m'a montré le robot me disait que le robot détecte la mammite qui rend le lait impropre à la consommation bien avant que lui-même, pourtant éleveur chevronné, ne saurait le faire.

Depuis quelques décennies, la tendance à l'automatisation et à la robotisation s'est considérablement accentuée et il n'est guère de machine utilisée aujourd'hui qui ne contienne un grand nombre de puces électroniques aux quelles sont confiées diverses opérations, sans que d'ailleurs l'utilisateur humain de la dite machine sache précisément les quelles. Une voiture automobile ordinaire en contient plusieurs dizaines : chacun sait que nos voitures actuelles freinent beaucoup mieux que les anciennes et que les chances de faire un tonneau avant de finir sur le toit dans un champ après avoir donné un coup de frein ont considérablement diminué. Ceci a été obtenu par un système, désormais entièrement électronique, qui s'insère entre la pédale de frein et les mâchoires, ni la pédale ni les mâchoires n'ayant beaucoup changé : la transmission de l'ordre de freiner, plus ou moins brutalement selon la force appliquée, qui est donné par le conducteur en appuyant sur la pédale ne se fait plus mécaniquement par un système de tringles. L'ordre est reçu et analysé comme une information qui est mélangée à d'autres sur la position de la voiture et transformé en le déclenchement d'une série de petits coups de freins très rapprochés les uns des autres et inégalement répartis sur les quatre mâchoires en fonction de la trajectoire connue de la machine. Le résultat de tout cela est que la proportion de puces électroniques qui servent à la fabrication d'ordinateurs est inférieure au dixième de la production totale, les autres, c'est-à-dire l'immense majorité se retrouve dans les machines les plus diverses allant des machines à laver et des jouets pour enfants aux centrales nucléaires et aux avions.

Ce qui s'est passé c'est que vers 1960 les algorithmes et les programmes sont devenus des objets d'études scientifiques. En fait ils l'étaient déjà depuis le début du vingtième siècle entre les mains d'une pléiade de logiciens mathématiciens parmi lesquels il faut au moins citer Hilbert, Gödel, Church, Kleene, Markov et Turing qui ont fondé la théorie du calcul. Mais ce qui a déclenché le vaste mouvement de recherche qui a abouti à l'informatique que nous connaissons aujourd'hui c'est l'apparition des langages symboliques de programmation, que nous disions à l'époque « de haut niveau ». Ils sont trois à avoir vu le jour en 1960, Fortran, Algol et Lisp. La notion d'algorithme était déjà bien connue, en particulier des mathématiciens : l'algorithme d'Euclide a deux mille cinq cents ans et le mot d' « algorithme » lui-même, forgé sur le nom du savant ouzbek Al Khorezmi qui vivait au huitième siècle de notre ère est attesté depuis le treizième siècle. Mais le langage mathématique, pas plus que le langage vernaculaire, n'était bien adapté à

la description d'algorithmes ni à l'écriture de programmes. Les premiers « langages machines » qui servaient à programmer les machines mécanographiques, nombreuses déjà après leur invention par Hollerith vers 1880, et les premiers ordinateurs qui en 1960 étaient encore très rares et très peu puissants étaient vraiment incommodes et surtout très différents selon les machines. Les trois langages que je viens de nommer ont permis à tout un tas de gens d'écrire des programmes qu'ils pouvaient échanger ; et accumuler : des mathématiciens ont repris et très vite amélioré les algorithmes mathématiques portant sur des nombres ou des figures géométriques ce qui a donné naissance à ce que nous appelons en France l'analyse numérique. Mais beaucoup de gens se sont penchés sur des algorithmes non numériques en particulier les algorithmes de rangement, de tri, de constitution de tables et de recherche dans des tables, de traitement de textes aussi. Ils ont élaboré des méthodes générales de construction d'algorithme, de mesure et d'accélération de leurs performances, de traduction d'un programme dans un autre, de vérification de la validité d'un programme (valide s'il fait bien ce pour quoi il a été conçu). Et ce sont ces méthodes qui constituent le cœur de la science informatique qui a permis, en même temps que les progrès de l'électronique, le formidable essor de l'informatique, comme outil et comme champ de la connaissance. Que la constitution de cette science ait incorporé une grande partie de savoir ancien voire très ancien est particulièrement apparent dans l'œuvre en trois volumes de Donald Knuth « the art of programming » qui est demeuré très longtemps la bible des informaticiens et il n'est pas inutile de savoir que le même Knuth a passé dix ans de sa vie à imaginer et mettre au point le langage TEX destiné à écrire et imprimer des textes mathématiques, y compris bien sûr les formules étranges qui peuvent les émailler.

Je crois qu'il est nécessaire ici de bien préciser que les avancées qui ont permis la constitution de la science informatique ne sont pas le fait des seuls informaticiens, l'informatique incorpore non seulement beaucoup de savoirs anciens mais aussi beaucoup de savoirs issus d'autres sciences ou d'autres techniques ou de domaines de savoir qui ne sont pas considérés comme très scientifiques. Les chimistes ont développé leur propre façon de représenter leurs molécules sur un écran, les architectes leur propre façon de transformer leurs croquis en dessins précis, les carrossiers leurs propres façon de dessiner les carrosseries de voiture et les fuselages d'avion, les linguistes ont développé des méthodes de traduction et d'analyse de textes, les médecins ont mis au point des techniques incroyables d'imagerie médicale. Mais tout cela se fait dans un cadre bien précis : la recherche de l'information nécessaire, la recherche de la représentation de cette information

la mieux adaptée au traitement qu'on souhaite lui faire subir, l'invention d'algorithmes et l'écriture de programmes pour effectuer ce traitement, la vérification que tout marche bien.

3. Réseaux et systèmes complexes

Je n'ai guère parlé jusqu'à présent que d'un opérateur, d'une machine, d'un outil cherchant à réaliser quelque chose, mais l'homme comme chacun sait est un animal social et la plupart des actions sont le fait d'un groupe ou d'une équipe. Un atelier ou une usine fait coopérer un certain nombre d'acteurs humains et de machines à la poursuite d'un certain but. Des ateliers de taille de pierres ou de poterie existaient il y a très longtemps et des solutions avaient été trouvées pour gérer ces ateliers, assignation des tâches à effectuer aux divers membres de l'atelier, circulation des pièces à divers stades de leur fabrication, utilisation des ressources, outils et matières premières. Une telle gestion soulève des problèmes de communication entre les membres de l'atelier, de coopération pour effectuer des tâches qui requièrent l'intervention de plusieurs d'entre eux, de concurrence pour synchroniser les actions. Communication, coopération, concurrence (en abrégé C cube, C^3) était le nom d'un groupe de recherche du CNRS qui a vécu d'environ 1980 à environ 1995, période pendant laquelle l'algorithmique et la programmation ont étendu leur champ pour traiter de problèmes qui font intervenir plusieurs acteurs et dans lesquels le temps intervient de façon cruciale. C'est à cette même période que sont apparus et se sont développé les réseaux dont Internet est le plus connu.

Les problèmes rencontrés ne sont pas les mêmes que ceux que rencontre la programmation séquentielle d'une seule machine effectuant des opérations les unes après les autres, ce qui évite tout conflit ; éviter les conflits est quelque chose à quoi nous sommes tous habitués depuis notre plus jeune âge, ne serait-ce que pour nous déplacer à pied, en vélo ou en voiture. Deux mobiles qui occupent le même emplacement au même instant se rentrent l'un dans l'autre avec les résultats plus ou moins désastreux que l'on sait. Si un agent de police ou un feu tricolore n'est pas là pour dire à chacun de ces mobiles quand il peut passer et arrêter ceux qui pourraient rentrer en collision avec lui, il faut qu'une information soit échangée entre les deux mobiles et établir des règles connues des deux mobiles disant quel est celui qui doit passer en premier. Le code de la route est là pour ça, l'information est visuelle ou sonore (klaxon) et en principe il ne doit pas y avoir de collision.

Les systèmes que nous appelons « temps réel » sont extrêmement répandus allant du restaurant de quartier à une centrale nucléaire ou un gros avion de ligne dans

lesquels il s'agit de décider d'agir le plus rapidement possible et de faire effectuer une certain nombre de tâches, par un certain nombre d'éléments du système en fonction d'informations arrivant en permanence sur l'état du système et de son environnement et cela a suscité une masse considérable de travaux. La mise au point et la vérification des logiciels qui assurent la bonne marche de nos avions avec une extraordinaire fiabilité ont été les résultats les plus spectaculaires de ces travaux. Et ne nous y trompons pas, leur complexité fait que l'écriture de ces logiciels est complètement sortie de l'artisanat, leur conception, leur écriture et leur vérification ne peuvent plus être envisagé qu'en utilisant des outils informatiques eux-mêmes fort sophistiqués et reposant sur des résultats scientifiques tout à fait nouveaux (par exemple le « model checking » outil essentiel aussi de la conception, du dessin et de la vérification des circuits intégrés de plusieurs millions d'équivalents-transistors sur lesquels tournent en permanence les logiciels en question). Il s'est révélé que la notion d'automate, telle qu'elle est apparue en 1956 sous la plume de Stephen Coole Kleene a joué un rôle fondamental dans cette conquête du temps réel par la programmation et dans la possibilité de dessiner et produire des circuits intégrés de la taille gigantesque qu'ils atteignent aujourd'hui. A l'instar des objets que manipulent les mathématiciens, ces automates jouent un double rôle conceptuel, au niveau de la conception, comme modèle de ce que l'on veut faire, et un rôle utilitaire en permettant calculs et vérifications (démonstrations de validité). Une fois encore l'histoire de ce concept est très instructive, le grand inventeur du dix-huitième siècle Jacques de Vaucanson qui outre construire son célèbre canard a fait faire des progrès substantiels aux métiers à tisser, en avait une idée encore bien floue bien que les automates qui l'ont rendu célèbre soient bien en effet des automates au sens moderne du terme. L'algèbre de Boole a rendu possible l'avènement de la mécanographie inventée par Hollerith et des machines, tabulatrices et trieuses, qui ont fait la fortune des compagnies IBM (fondée initialement par Hollerith lui-même) et Bull : ces machines sont des automates dont les transitions sont réalisées au moyen de portes logiques composées de lampes diodes. Et toute une théorie dite en anglais « switching theory » (théorie de la commutation) s'était développée et était enseignée à Berkeley et autres lieux dans des départements d' « electrical engineering » encore en 1960 : ce qui était de fait une théorie des automates, qui servait à la mécanographie mais aussi à la fabrication de tous les servomécanismes et des autocommutateurs téléphoniques a disparu au profit d'une théorie des automates, dans la définition qu'en donne Kleene, qui s'est considérablement enrichie depuis et constitue un chapitre essentiel de l'informatique.

Avec un peu de retard, la programmation temps réel et multi-acteurs est en train d'envahir le monde de l'entreprise sous la forme de systèmes d'information. Les entreprises ont commencé très tôt à utiliser des techniques informatiques pour gérer les informations qu'elles doivent gérer, informations sur le personnel, sur les fournisseurs, sur les clients, sur les stocks de matières premières, sur les stocks de produits manufacturés, sur les commandes etc.… Et elles ont été avec les administrations les plus grands producteurs et consommateurs de bases de données plus ou moins structurées. Tout cela ne touchait pourtant pas le cœur de l'entreprise, n'affectait pas sa stratégie. Les systèmes d'information vont beaucoup plus loin car ils gèrent aussi la circulation de toute l'information qui est produite par l'entreprise entre les divers éléments, en s'assurant que chacun reçoit toutes les informations qui lui sont nécessaires et celles-là seulement. Parmi ces informations il y a évidemment celles qui disent qui fait quoi à chaque instant et le liste des tâches qui restent à accomplir constamment mise à jour. Ainsi connaissant tout ce qui se passe à chaque instant dans toute l'entreprise, le système peut faire beaucoup de choses : s'assurer que chacun exécute bien dans le temps imparti les tâches qui lui ont été confiés, éviter que des taches soient exécutées plusieurs fois ou par des gens qui n'ont pas vocation à les exécuter, optimiser (selon des critères à définir et modulables) l'assignation des tâches aux divers exécutants, détecter tous les dysfonctionnements, retards, anomalies et y remédier par une modification du programme de travail de certains éléments. Le système devient le poumon de l'entreprise en même temps que son cœur tant il est vrai que c'est l'information dont dispose chacun qui détermine la bonne marche de l'ensemble. Les militaires, qui ont eux aussi, des actions collectives à piloter ont précédé les civils dans la mise au point de système d'information qu'ils appellent systèmes d'armes.

4. Quelques considérations sur la science en général et ses rapports à la technique

A travers une meilleure connaissance de l'information et des moyens de la recueillir, de la représenter de la diffuser et de l'utiliser, et par une réflexion sur les processus algorithmique dans tous les domaines, l'informatique est devenue une science qui comme toutes les autres reposent sur des concepts, sur l'observation de phénomènes, et sur des méthodes applicables dans le plus grand nombre de situations. C'est aussi la science de l'outil, la science de la machine et la science de l'action individuelle ou collective. En cela elle ne se distingue pas des autres sciences dites exactes, mathématiques, physiques, chimie, biologie.

Mais c'est quand même une science singulière car son champ englobe la quasi-totalité des connaissances et des activités humaines.

Toutes les connaissances accumulées sont des informations dont les conditions d'utilisation, à quelque fin que ce soit, sont profondément modifiées par l'informatique plus encore sans doute qu'elles n'avaient été modifiées à la Renaissance par l'apparition du livre.

Toutes les activités sont algorithmiques, des plus triviales aux plus complexes et sophistiquée si bien qu'elles sont toutes susceptibles d'être revues et corrigées à la lumière des connaissances nouvelles acquises sur les algorithmes devenus objets d'études nombreuses. Et c'est bien ce qui se passe depuis plusieurs décennies, les puces viennent rendre les outils et les machines de toutes espèces plus performants ou plus fiables, dans toutes les chaînes mélangeant actions humaines et actions de machines. La part de l'homme diminue et celle de la machine augmente. Des méthodes nouvelles de coopération et de communication entre acteurs humains éventuellement très nombreux sont apparues entraînant des modifications importantes des comportements et des possibilités d'actions collectives nouvelles (là nous n'en sommes qu'au début, le cloud computing, et l'intelligence collective n'ont pas fini de faire parler d'eux!).

Il va de soi que l'omniprésence des algorithmes et des programmes est telle que la corporation des informaticiens ne peut prétendre à les connaître tous ni à être chargée de leur entretien ou surveillance. Chaque jour des algorithmes nouveaux sont inventés et proposées par quantités de gens qui ne sont pas des informaticiens et dont même souvent les compétences informatiques et la connaissance de la science informatique sont faibles sinon nulles. Et ceci engendre bien des malentendus: à quoi sert la science informatique si moi, qui n'y connaît rien je peux inventer un algorithme et même, tant bien que mal, le programmer?

Les informaticiens ou du moins les marchands de services et systèmes informatiques font à cette question deux réponses parfaitement contradictoires:

-des sociétés de service et les marchands de gros ordinateurs dit mainframe disent "laissez-nous faire, vous n'y comprendrai rien mais ça marchera".

-des marchands d'ordinateurs et de logiciels disent "tout est très simple, vous n'avez besoin de rien savoir pour utiliser nos matériels et nos logiciels".

Les premiers vendent atrocement cher des services et des matériels en relativement faible quantité, les seconds vendent relativement bon marché mais en très grande quantité des logiciels dans les supermarchés et il est clair que ces deux réponses sont des arguments publicitaires également mensongers.

La science informatique sert à ce quoi servent toutes les sciences, accumuler des connaissances, les structurer, les rendre plus facilement assimilables, accompagner le progrès technique qui parfois la précède et parfois en découle. La différence entre science et technique est beaucoup plus sociale, voire caractérielle, que fondamentale : par goût ou par nécessité des gens préfèrent faire, deviennent ingénieurs ou techniciens, d'autres, je pense surtout par goût, préfèrent comprendre et expliquer. Et la différence peut être extrêmement ténue : l'ingénieur qui fait du développement pointu n'a pas une vie bien différente du chercheur scientifique. L'un utilise toutes ses connaissances dont la plupart sont scientifiques pour fabriquer un objet ou un système, l'autre utilise toutes ses connaissances dont certaines doivent être techniques pour faire avancer la connaissance. L'incroyable progrès technique qu'a connu l'informatique et qui continue, fait que la différence entre technique et science est encore plus difficile à discerner en informatique qu'en mathématique par exemple. Du théorème, qui s'il n'est pas trivial peut avoir nécessité un temps assez long de démonstration, à son éventuelle application il y a en général encore un grand pas à franchir, de l'idée scientifique d'un informaticien à son utilisation il n'y a souvent aucun pas à franchir, la plupart des idées se traduisent directement par des augmentations de performance d'algorithmes et programmes existants qui sont validées directement par une implémentation ou la réalisation d'un système expérimental. La proximité de la technique, toujours très proche de l'informaticien, a une autre conséquence. Comme le disait André-Georges Haudricourt, la technique est toujours un peu une science humaine, tout ce qu'elle construit est destiné à être utilisé par des hommes et de femmes et le plus bel objet technique, le système le plus magnifique ne sert à rien s'il est trop compliqué ou malcommode de s'en servir. Les informaticiens sont, bon gré mal gré, mais pour la plupart par conviction, moins enclins que d'autres scientifiques à se lancer dans des théories absconses et destinées aux seuls « happpy few », peu nombreux qui peuvent les comprendre.

Pour de nombreuses raisons je pense que la science informatique ne prendra jamais la forme très centralisée des mathématiques ou chaque théorème trouve sa place dans ce que les mathématiciens aiment à voir comme la plus belle construction de l'esprit, construction que tout mathématicien est censé pouvoir reconstruire par lui-même et à la seule force de son raisonnement. L'informatique est trop diverse, se rencontre en trop de circonstances qui n'ont rien à voir les unes avec les autres (en apparence du moins) pour qu'un informaticien l'embrasse tout entière, il y aurait plutôt des informatiques qui se développeraient en liaison avec différents grands domaines d'activité et de connaissances et qui ne seraient reliés entre eux que par un ensemble commun d'outils conceptuels et de grandes méthodes générales. En

dehors des lois de la physique, et des théorèmes des mathématiciens il y a de grands principes qui ont fait avancer considérablement les sciences, on connaît le principe de Carnot en thermodynamique et la méthode de Descartes qui sous-tend la plus grande partie de l'activité en sciences exactes consistant à aller toujours du particulier au général ; Un résultat n'est jamais que l'extension d'une propriété observée dans des cas particulier à un cas général qui englobe de nombreux cas particuliers.

L'informatique avance aussi bien en tant que science qu'en tant qu'outil grâce à une démarche universelle : chercher toute l'information nécessaire à une action complexe avant de commencer à faire quoique ce soit, chercher toutes les actions élémentaires qui seront nécessaires ou qui sont disponibles avant de se lancer dans la planification d'une action complexe. Les progrès de l'algorithmique, les gains de performance et d'efficacité sont en général obtenus par une réflexion approfondie sur l'information apportée par chaque opération élémentaire qui souvent apporte plus d'information, utilisable dans la suite du processus qu'on ne le croyait. La possibilité de concevoir et réaliser des systèmes et des programmes de plus en plus complexes est fourni par le fait que de plus en plus d'opérations en elle-même complexes peuvent être considérées comme élémentaires, une fois bien programmée, testée et vérifiée. Ainsi le stock de briques de base avec lequel on construit algorithmes et programmes s'enrichit constamment.

De la même façon que la notion de mesure d'une grandeur physique doit être acquise par tous, il me parait nécessaire que tous les citoyens, et certainement tous les jeunes soient confrontés et familiarisés avec la démarche informatique reposant sur quelques notions fondamentales concernant l'information et ses représentations, les algorithmes et les programmes. Non seulement la démarche informatique n'a pas apporté tout ce qu'elle peut apporter mais nous ne sommes qu'au début de l'informatisation du monde.