

# Enseigner les Systèmes Numériques en 2049

Eddie Smigiel

eddie.smigiel@insa-strasbourg.fr

Archives Henri Poincaré, UMR 7117

CNRS - Université de Lorraine - Université de Strasbourg,

INSA de Strasbourg,

24, Bd de la Victoire 67084 Strasbourg Cedex

François de Bertrand de Beuvron

francois.debertrandbeuvron@insa-strasbourg.fr

Laboratoire Icube, équipe SDC, UMR 7357

CNRS - Université de Strasbourg, ENGEES, INSA de Strasbourg,

24, Bd de la Victoire 67084 Strasbourg Cedex

**RESUME :** La complexité des systèmes numériques est devenue telle qu'il n'est plus possible de les enseigner exclusivement depuis les sciences fondamentales (mécanique quantique et physique statistique) vers les systèmes commerciaux réels, méthode qu'on peut qualifier de méthode ascendante. Ainsi, la méthode descendante est apparue depuis une vingtaine d'années, souvent enseignée en mode projet, qui consiste à utiliser un système commercial (une plate-forme de type Arduino ou Raspberry Pi pour ne citer que les plus populaires) dans le cadre d'un développement confié aux étudiants avec un cahier des charges donné. Dans cet article basé sur des considérations essentiellement épistémologiques, on met en évidence qu'il importe de trouver un équilibre entre les deux méthodes, équilibre qui ne va pas nécessairement de soi. Il existe un risque non négligeable, à terme, que les collectifs quels qu'ils soient perdent la maîtrise des systèmes numériques dont la complexité se rapproche de la complexité biologique ce qui modifierait la nature profonde du rapport à ces objets pourtant construits par l'Humanité.

**Mots clés :** électronique numérique, complexité, épistémologie.

## 1. INTRODUCTION

La communauté des Sciences de l'Ingénieur a vécu depuis une trentaine d'années un événement probablement unique dans l'Histoire de l'Humanité : l'émergence d'une complexité inédite dans les systèmes numériques susceptible de bouleverser de façon irréversible les méthodes d'enseignement. Quand les premiers microprocesseurs étaient constitués de quelques milliers de transistors, un processeur de smartphone possède aujourd'hui plusieurs milliards de transistors. Tout ce qui a été construit par l'Humanité jusqu'alors avait la caractéristique suivante : la construction procédait d'une chaîne visible de connaissances et techniques que l'honnête homme, au sens de Diderot, était capable d'appréhender dans ses principes à défaut d'en maîtriser toutes les subtilités. Une cathédrale ne peut être construite par un individu unique ni même une seule profession et il faut de nombreux corps de métier pour y parvenir. Aucun être humain n'est capable dans l'espace d'une vie d'acquérir la totalité des savoir-faire de ces corps de métier. Mais l'ensemble des tâches nécessaires peut être raisonnablement appréhendé dans ses principes par l'esprit humain tant et si bien que l'objet ne paraît pas surnaturel même si on reste songeur devant la monumentalité de certains édifices qui impressionne notre imaginaire par l'énormité des efforts auxquels il a fallu consentir pour les édifier. Il en va tout autrement du vivant.

Comprendre le dauphin ne peut dans l'état actuel des connaissances se faire depuis les sciences les plus fondamentales, mécanique quantique, par exemple. Quand bien même le dauphin est constitué des atomes de la table de Mendeleïev, atomes dont l'organisation dérive de la physique fondamentale, il y a loin de la physique théorique aux techniques de chasse du dauphin et à chaque transition de discipline scientifique, de la physique à la chimie, de la chimie à la biochimie et enfin, de la biochimie à la biologie, on constate une émergence de complexité telle que l'Humanité est bien incapable d'expliquer de façon cohérente et unifiée l'ensemble des causalités et interactions qui ont conduit à la biodiversité. Mais trois milliards d'année d'évolution sont passées par-là et on accepte aisément que cette complexité biologique ne puisse être approchée par une construction ascendante depuis la physique fondamentale vers l'émergence de la complexité du vivant. Pour ce qui concerne les systèmes numériques, force est de constater qu'ils sont à cet égard un peu hybrides. Ils ne sont pas êtres vivants, résultats d'une lente évolution. Ils sont construits par l'être humain. D'un autre côté, leur complexité commence à ressembler à celle d'un objet vivant, d'autant qu'ils commencent à incorporer de l'intelligence artificielle (IA). Bien que cette complexité n'ait pas encore atteint celle des systèmes biologiques avec leur corollaire de concepts d'émergence et d'auto-organisation (les milliards de transistors ne s'auto-organisent

pas spontanément mais sont placés très précisément sur un schéma par l'ingénieur), il devient néanmoins difficile pour quiconque de comprendre leur nature construite depuis la science fondamentale vers les objets opérationnels constitués de milliards de transistors qui font tourner des dizaines de millions de lignes de code. Des questions d'éthiques, que l'on croyait du seul domaine du vivant, et plus spécifiquement de l'humain commencent d'ailleurs à se poser concernant en particulier les matériels autonomes pilotés par IA [5].

Dans cet article qui pose plus de questions qu'il n'apporte de réponses, nous nous essayons à une réflexion philosophique ou autrement dit, épistémologique, avec toutefois quelques éléments très concrets sur quelques principes didactiques qui pourraient guider les pratiques d'enseignement de l'avenir. Nous essayons de dégager quelques enjeux, possibilités mais aussi risques pour anticiper modestement sur ce qui pourrait constituer un enseignement des systèmes numériques en 2050 (le 2049 du titre de l'article est un clin d'œil au film « Blade runner 2049 » réalisé par Denis Villeneuve en 2017).

## 2. ENSEIGNEMENT TRADITIONNEL

### 2.1. La méthode ascendante ou analytique

Les systèmes numériques sont construits par l'homme ex-nihilo et n'ont pas pour l'heure de capacités évolutives propres même si cette ère semble à portée de main au moins pour la partie logicielle, avec l'arrivée de l'Intelligence Artificielle. Si quiconque en tant qu'individu est largement dépassé par la somme de toutes les connaissances scientifiques et technologiques nécessaires pour les construire et poursuivre leur développement, il n'en reste pas moins que l'Humanité reste en mesure de le faire. Les microprocesseurs ne poussent pas sur les arbres. A ce titre, il est possible de les enseigner par une méthode qu'on peut qualifier d'ascendante ou analytique en ce sens qu'on part de sciences fondamentales pour aboutir aux connaissances et compétences propres au métier pour les réaliser sur un plan industriel. Toutes les relations de cause à effet sont descriptibles et maîtrisées à défaut d'un seul individu, par la communauté des ingénieurs. On s'efforce dans les lignes qui suivent d'esquisser la lignée des connaissances en jeu depuis les couches les plus fondamentales jusqu'aux niveaux opérationnels les plus actuels, par un modèle en quatre niveaux de description. On pourrait ajouter un cinquième niveau, celui de l'IA et des systèmes fondés sur les algorithmes d'apprentissage profond. Mais ceux-ci soulèvent des problématiques spécifiques liées à la perte de déterminisme qui dépassent le cadre de cet article.

Au commencement était la mécanique quantique qui régit le comportement des porteurs de charges électriques dans les semi-conducteurs, les électrons et les trous. Combinée essentiellement à la physique statistique qui régit le comportement d'un ensemble de particules, on arrive à la physique des solides et en tant que sous-ensemble, à la physique des semi-conducteurs. La théorie des bandes par exemple, si essentielle pour décrire l'état

énergétique des porteurs dans un semi-conducteur, s'obtient par la résolution de l'équation de Schrödinger dans un cristal périodique. On parvient alors à décrire les semi-conducteurs intrinsèques aux applications toutefois assez limitées. On peut ensuite poursuivre par le phénomène de dopage qui conduit aux semi-conducteurs extrinsèques puis rapidement à la jonction PN. Les éléments qui précèdent, qu'on peut définir comme un premier niveau de description, décrit notamment dans [1], sont du domaine exclusif de la physique. Mais lorsqu'on arrive à la jonction PN, on parvient à un phénomène qu'on peut comparer à une transition de phase au sens didactique. La jonction PN conduit immédiatement à la diode à jonction qu'on peut appréhender comme objet du « Génie électrique » avec son épistémologie propre quand on envisage le dipôle dans un circuit fonctionnel (par exemple, un redresseur). Cette transition de phase correspond à une rupture de la continuité didactique en ce sens que la caractéristique courant – tension de la diode, par exemple, peut être utilisée de manière axiomatique sans qu'il soit indispensable d'enseigner les éléments amont, mécanique quantique et physique des semi-conducteurs. De toute manière, le concept de diode existait avant l'apparition de la technologie à base de semi-conducteurs quand elles étaient réalisées avec la technologie des tubes électroniques à vide. Nous arrivons alors après cette transition de phase, à un second niveau de description qui va de la jonction PN jusqu'au transistor dont la variante transistor bipolaire consiste en une double jonction NPN, par exemple. Ce deuxième niveau de description est hybride en ce sens qu'il peut être abordé avec l'œil du physicien ou au contraire celui du génie électrique. Il peut être appréhendé avec un ensemble axiomatique adéquat sans avoir étudié en détail le premier. Les deux disciplines, physique et Génie électrique ne correspondent pas aux mêmes problématiques, ni aux mêmes enjeux, de façon générale, à la même épistémologie. Historiquement, les cursus « physique appliquée » et « Génie électrique » de l'École Normale Supérieure de Cachan et les agrégations éponymes maintenant disparus ou fortement remaniés illustrent bien à la fois la proximité et aussi la subtile différence entre l'épistémologie de la physique et celle du Génie électrique. Quand on domine bien le composant transistor, on arrive alors au concept de porte logique élémentaire, ce qui constitue une deuxième transition de phase et le début d'un troisième niveau de description. En effet, avec quelques transistors utilisés en saturation (qui se comportent donc comme des interrupteurs commandés), on réalise des portes logiques élémentaires, d'abord combinatoires (NON, ET, OU, etc.) puis séquentielles (bistables RS puis bascules). Ce troisième niveau de description, décrit notamment dans [2], est exclusif des « Génie électrique ». Les physiciens ne sont plus concernés. Dans ce troisième niveau de description d'où la physique est absente, on trouve entre autres, et outre les portes élémentaires déjà évoquées, la synthèse des fonctions combinatoires, l'algèbre de

Boole, les systèmes de numérations binaires, les mémoires, les compteurs et registres qui aboutissent ensuite au concept de machine à état. Enfin, avec l'Unité Arithmétique et Logique (un circuit combinatoire) et le séquenceur qui est une machine à états, on peut introduire le concept de microprocesseur de type Von Neumann. On renvoie notamment à [2], une série de manuels assez synthétique et complet de ce niveau de description. Dans ce troisième niveau de description, on a aussi introduit la notion de jeu d'instructions d'un microprocesseur simplifié et donc au langage machine et éventuellement au langage assembleur. Avec le langage assembleur, on assiste à une première tentative d'abstraction qui consiste à s'émanciper du niveau de description binaire du jeu d'instruction. En effet, le langage assembleur est un intermédiaire entre le niveau profond où une instruction est décrite par un octet ou plus généralement un mot de  $n$  bits et une description en langage pseudo-naturelle mais tout en restant très proche du jeu d'instructions. Dans le niveau d'abstraction suivant qui consiste à invisibiliser les couches profondes, on pose le concept de langage compilé comme le C, par exemple, qui se rapproche d'un langage naturel. On permet ainsi à des non « Génie électrique » d'accéder à la programmation. Dans la transition de phase langage assembleur vers langage compilé, on débouche sur le quatrième et dernier

niveau de description de notre modèle, celui de l'informatique ainsi fondée en discipline autonome voire indépendante avec son épistémologie propre. L'informatique moderne a à ce point invisibilisé les couches profondes qu'on peut l'enseigner en tant que telle sans avoir étudié un tant soit peu les niveaux de description précédents. De manière générale, chaque niveau de description peut correspondre à une formation voire un diplôme sans qu'il soit indispensable de les combiner. Néanmoins, dans bien des formations d'écoles d'ingénieurs, on ne se contente pas d'enseigner un niveau de description hors sol mais généralement, on les combine dans une infinité de variantes en termes de volumes horaires ou crédits alloués avec évidemment une problématique d'optimisation dans le contexte de la multiplication des connaissances, des moyens au mieux constants et de la demande toujours croissante de disciplines nouvelles dans les formations d'ingénieurs notamment. La figure 1 représente ce modèle en quatre niveaux de description. On y a ajouté en pointillé, à la droite de l'informatique, un bloc « Intelligence Artificielle » avec ses propriétés et problématiques propres. Les auteurs ont choisi de placer cette transition de phase en dehors du champ du présent article.

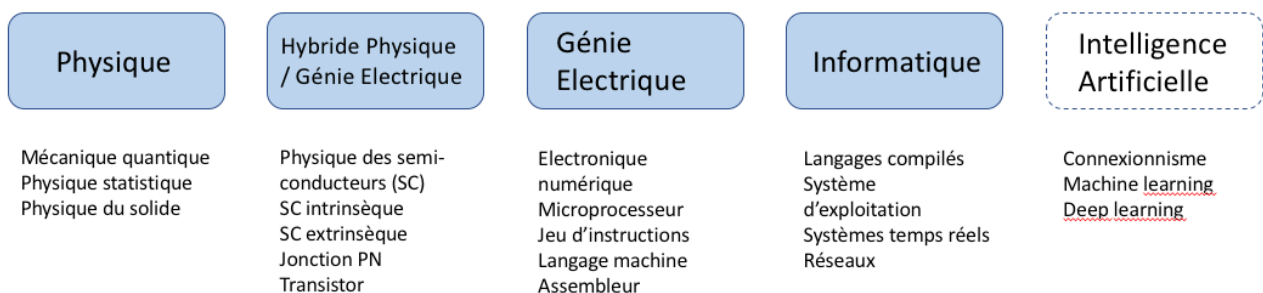


fig 1 : Modèle en quatre niveaux de description (le cinquième est évoqué en pointillé) d'enseignement de l'Électronique numérique depuis les sciences fondamentales jusqu'à la réalisation pratique d'un système opérationnel

## 2.2. Les limites de la méthode

Dans la méthode décrite précédemment, on observe une forme de continuum depuis les connaissances les plus fondamentales jusqu'à l'objet ultime, le système numérique à logique programmée sans que jamais la chaîne didactique ne soit rompue. Au moins sur le plan qualitatif, l'étudiant a des réponses de principe sur le fonctionnement de l'objet ultime sans que rien ne soit mystérieux. Dès lors qu'on sait fabriquer un transistor, il ne reste plus qu'à les assembler selon les principes de l'électronique numérique, et on est capable, en principe, de réaliser un système numérique au sens hardware ou matériel ainsi que le software ou code qui l'exploite. C'est une approche qu'on peut qualifier d'exhaustive, en ce sens que rien n'est admis ou donné dont le fonctionnement serait opaque voire magique. C'est donc une approche séduisante sur le plan philosophique. Elle a été

naturelle dans les années 1970 puis 1980 tant que la complexité technologique ne s'était pas trop émancipée des principes, concrètement, tant que le nombre de transistors restait relativement modeste. A partir des années 1990, la technologie s'est à ce point complexifiée et surtout, on a vu apparaître des systèmes tels que la connaissance de la chaîne exhaustive n'est plus indispensable. Depuis une dizaine d'années au moins, cette approche est remise en cause de fait. Dans tous les systèmes numériques, ordinateurs classiques, smartphones et tablettes ou plateformes dédiées aux fablabs de type Arduino, Raspberry Pi (pour ne citer que les plus connues) ou équivalents, il n'est plus guère nécessaire d'enseigner tous ces niveaux de description dans une optique purement opérationnelle. L'étudiant, notamment en projet, devient capable de mettre en œuvre ces systèmes sans pour autant maîtriser les quatre niveaux de description

et cela même dans une modalité très simplifiée. A la limite, on peut développer sur les plateformes dédiées aux fablabs qui permettent de mettre en œuvre tous types d'entrées / sorties pour interagir avec l'extérieur sans rien connaître des trois premiers niveaux de description. Ces systèmes sont d'ailleurs conçus pour qu'il en soit ainsi et tout est mis en œuvre pour que tout un chacun soit en capacité de les utiliser, notamment une très abondante documentation en ligne et des bibliothèques de projet finalisés, tant et si bien que l'étudiant, au sens large, procède souvent par de légères modifications de projets existants, quelquefois uniquement de la partie code. La méthode traditionnelle est dès lors remise en cause jusque dans ses fondements. L'idée intuitive selon laquelle la connaissance exhaustive au moins des principes est utile peut même être contestée et on peut se poser la question de savoir si la satisfaction intellectuelle de comprendre en quoi un objet numérique est construit et connu entièrement confère un avantage quelconque lorsqu'on veut les utiliser dans un projet. Clairement, la méthode traditionnelle ascendante ou analytique consomme beaucoup de temps, met en lumière des connaissances et techniques qui peuvent sembler datées (par exemple du bistable à la bascule) et l'étudiant moyen peut s'impatienter puisqu'il est établi aujourd'hui qu'il veut des activités en capacité de l'impliquer et de l'enthousiasmer. Ainsi la méthode traditionnelle est même susceptible de devenir contre-productive à l'instar d'un élève de conservatoire que l'étude prolongée du solfège finit par ennuyer alors qu'il rêve de pratiquer un instrument.

### 3. ENSEIGNEMENT MODERNE

#### 3.1. La méthode descendante ou systémique

De fait, on constate que dans de nombreux cursus, on enseigne ces disciplines souvent dans le cadre des objets connectés essentiellement par le mode projet dans lequel les étudiants pratiquent jusqu'à la réalisation d'un système opérationnel. A la limite, le projet peut être conduit uniquement de manière logicielle avec un langage compilé ou interprété mais toujours haut niveau et il est devenu rarissime que les étudiants soient confrontés à des codes en binaire ou hexadécimal. Comme dans le même temps, le projet fait souvent usage de technologies très modernes et partant très complexes, comme une liaison Bluetooth, par exemple, il semble évident que si on souhaite amener les étudiants à la capacité de mettre en œuvre, il faut renoncer à la méthode ascendante dans laquelle toute la chaîne de connaissances et technologies est visible au moins dans ses principes. On peut ajouter que ce type d'enseignements est déjà en vigueur depuis plusieurs années au lycée, notamment au bac STI2D. Dans les lignes qui suivent, nous nous efforçons de décrire un tel enseignement qu'on peut qualifier de descendant ou systémique en ce sens qu'il part d'une plateforme existante et qu'il « descend » vers une réalisation fonctionnelle exécutée par l'étudiant. En principe, la méthode « descend » également vers des niveaux de

description plus profonds mais il ne semble pas aller de soi que par la pratique d'un projet, on aboutisse à coup sûr à des connaissances historiques en électronique numérique (par exemple, l'architecture d'un microprocesseur).

D'un point de vue pratique, dans l'enseignement de type descendant, souvent pratiqué en mode projet, on réalise souvent une analyse fonctionnelle pour aboutir à l'inventaire des fonctions primaires et secondaires. Dans le cadre d'un système numérique, au sens large, cela revient généralement à faire le bilan des entrées / sorties, de l'alimentation, des parties opératives et de l'interface Homme / Machine, entre autres. Le choix de la plateforme est souvent relativement arbitraire compte tenu de l'offre très vaste. Parallèlement, si le groupe projet fait un travail sérieux d'étude de l'existant, il est probable qu'il identifiera des projets existants aboutis et publiés, qui sont proches. Dans certains cas, relativement nombreux, on pourra trouver des projets répertoriés dans le grand entrepôt Internet, des réalisations qui correspondent à des travaux finis exploitables moyennant à la limite de petites adaptations. L'offre étant très riche sur le plan hardware, le projet consiste souvent à acquérir les cartes électroniques filles (on parle souvent de shield) qui complètent la plate-forme avec son microprocesseur et on peut télécharger des codes très riches et complets qui permettent de les mettre en œuvre. Une difficulté consiste alors à identifier les parties de code qui doivent être implémentées ou simplement modifiées. Mais la maîtrise complète de la chaîne n'est plus indispensable.

#### 3.2. Avantages et inconvénients de la méthode descendante

Les avantages de ce type d'approches ne sont plus à démontrer. L'étudiant est acteur dans des activités qui généralement suscitent enthousiasme et désir d'aboutir à quelque chose qui fonctionne. Souvent, les étudiants ne comptent pas leurs heures et font preuve d'esprit d'initiative, d'endurance et de ténacité. Cette méthode favorise aussi les approches collaboratives, le management de projet. Pourtant, la méthode comporte aussi quelques inconvénients qu'on s'efforce de mettre en évidence dans les lignes qui suivent.

En premier lieu, les étudiants ont tendance à « deviner » les réponses aux questions que pose le système étudié. D'une part parce que le système n'est plus appréhendé comme étant entièrement construit et qu'on a donc perdu dans une certaine mesure la chaîne des causalités qui permet de répondre sur la base de la théorie et d'autre part en raison de la disponibilité totale des plateformes sur lesquelles ils travaillent. Pour illustrer ce dernier point, rappelons qu'au milieu des années 80, pour implémenter le code dans l'UVPRM d'une carte mère, il fallait compiler un code, puis éditer les liens et ensuite programmer la ROM sur un programmeur avant de l'insérer sur la carte mère. Les capacités de simulation étaient limitées et on y réfléchissait à deux fois avant de lancer une modification qui pouvait prendre plusieurs heures. Il était donc nécessaire de théoriser beaucoup.

Pour économiser un temps précieux, les programmeurs chevronnés écrivaient quelquefois directement dans l'UVPRM les codes en hexadécimal ce qui supposait une connaissance pointue du jeu d'instructions et des modes d'adressage du microprocesseur que seule la méthode ascendante pouvait enseigner. Aujourd'hui, ces niveaux de description sont totalement invisibilisés et les plateformes sont munies de capacités énormes de simulation en ligne telles que par essais erreurs successifs, on parvient à « deviner » une bonne réponse sans parfois avoir forcément compris le bien-fondé de la réponse. La disponibilité et la flexibilité des systèmes a donc aussi un revers de médaille, une certaine perte d'expertise qui conduit certains utilisateurs à les utiliser sans beaucoup de regard critique. On peut alors aboutir à quelque chose qui fonctionne - et l'étudiant se contente souvent d'en arriver là - mais qui ne serait pas réalisé dans les règles de l'art, et pourrait facilement aboutir à un système peu efficace, et à des erreurs : le but d'un ingénieur n'est pas uniquement de faire fonctionner un système. Il consiste aussi à l'optimiser, à qualifier sa sûreté et fiabilité, à le faire évoluer, etc. La méconnaissance des couches inférieures prive ainsi le développeur du regard critique sur la zone de validité et sur les performances de sa solution. Dans une approche descendante dans laquelle on a perdu une grande partie des causalités qui précisément permettent d'accéder à ces niveaux de descriptions, on risque d'être assez vite coincé à un constat trivial « ça marche » versus « ça ne marche pas » et donc de ne pas savoir répondre à la question : si je modifie légèrement les conditions, est-ce que cela « marchera » encore ?

Sur un plan plus philosophique, et à plus ou moins long terme, l'objet pourtant construit par le génie humain, risque d'être appréhendé comme un objet un peu fétiche dont il faut renoncer à expliquer le fonctionnement par méthode ascendante et dont on est réduit à observer le comportement dans l'espoir de le dompter. Mais à long terme, il y a un risque certain de perte de rationalité dans l'activité de l'ingénieur qui devient un peu sorcier en ce sens que sa longue expérience lui a permis d'accumuler des recettes et remèdes qui toutefois s'apparentent quelque peu à l'art divinatoire.

Sur un plan plus philosophique, l'artefact construit par le génie humain, risque d'être appréhendé comme un objet fétiche dont il faut renoncer à expliquer le fonctionnement et dont on est réduit à observer le comportement dans l'espoir de le dompter. À plus long terme, il y a un risque certain de perte de rationalité dans l'activité de l'ingénieur qui devient un peu sorcier en ce sens que sa longue expérience lui a permis d'accumuler des recettes et remèdes qu'il applique sans plus en comprendre la conception.

### 3.3. Bonnes pratiques possibles

Pourtant ce tropisme, mouvement vers des objets de plus en plus complexes, dont on invisibilise la nature profonde au profit d'une approche fonctionnelle de type « boîte noire » est apparemment inéluctable et il faut

bien s'en accommoder. Dans les quelques lignes qui suivent, on s'efforce de proposer quelques modalités simples pour trouver un équilibre entre les deux approches, ascendante et descendante, pour obtenir une sorte d'optimum entre expertise de la nature profonde des objets et compétences opérationnelles de l'ingénieur fraîchement formé. A trop former de façon ascendante, on privilégie une construction intellectuelle très complète, un peu encyclopédique mais sans certitude que ces connaissances ne seront jamais utiles dans la carrière, tandis qu'à trop former en méthode descendante, on risque d'obtenir des experts « débrouillards » et entrepreneurs mais dont les fondements scientifiques sont fragiles ce qui obère leur adaptabilité en cas de rupture technologique majeure.

Il semble évident que compte tenu de l'explosion des technologies, il n'est plus possible de traiter l'ensemble des contenus par la méthode ascendante. Si on prend l'exemple d'un objet communicant qui implémente du machine learning, il faudrait de très longues années pour former par méthode ascendante un jeune ingénieur spécialiste tout à la fois des systèmes de télécommunication, du machine learning et des systèmes numériques de façon générale pour obtenir les niveaux d'expertise les plus pointus dans chacune de ces disciplines. Aujourd'hui, ça semble impossible. Il faut dès lors admettre que l'étudiant mette en œuvre des connaissances et techniques qu'il ne maîtrise plus en totalité. Il faut donc parvenir à une formation équilibrée, par un savant mélange de méthode ascendante et descendante pour chaque élément du spectre des contenus qu'on souhaite transmettre.

Une première méthode consiste à s'en remettre à la sagesse spontanée des organisations de formations qui feraient ce savant mélange de façon un peu intuitive. Il est assez raisonnable de penser que c'est effectivement ce qui se passe aujourd'hui dans biens des formations de l'enseignement supérieur. Mais peut-être est-ce dû en partie au fait qu'il reste dans ces formations des professeurs formés « à l'ancienne », par méthode ascendante. La question se poserait alors pour l'avenir quand ces enseignants auront quitté leurs fonctions.

En second lieu, ces enjeux sont intrinsèquement traités, au moins en partie, par les approches « compétences » qui continuent de se développer sous l'impulsion des agences d'accréditation comme la CTI (Commission du Titre d'Ingénieur), par exemple. Toutefois, les approches des différentes agences d'accréditation et surtout leurs déclinaisons dans les formations d'ingénieur ne semblent pas à ce point unifiées ou matures qu'elles permettraient de prescrire un savant mélange susceptible de trouver un optimum raisonnable entre les injonctions contradictoires. Pour préciser un peu ce point, il est utile de regarder la classification des « learning outcomes » du cadre Eur-Ace du European Network for Accreditation of Engineering Education :

- Knowledge and understanding;
- Engineering Analysis;

- Engineering Design;
- Investigations;
- Engineering Practice;
- Making Judgements;
- Communication and Team-working;
- Lifelong Learning.

Les méthodes ascendantes relèvent clairement des trois premières catégories alors que la méthode descendante relève essentiellement du « engineering practice ». La catégorisation Eur-Ace semble donc pertinente pour formaliser la problématique de cet article. Néanmoins, la description d'un cursus entier dans ces niveaux de description jusque dans les pratiques d'évaluation des enseignants est un chantier dont tous les établissements se sont emparés mais qui n'est certainement pas encore totalement abouti.

Les auteurs ne disposent pas de solution toute faite. En revanche, ils sont convaincus que la question se doit d'être posée pour prévenir le cas échéant une lente dérive des formations qui seraient tentées de tout traiter par la méthode descendante, notamment en mode projet.

#### 4. UN ENJEU DE SOCIÉTÉ

Pour prolonger un peu le thème et pour esquisser une perspective plus large sans pour autant la traiter sérieusement, on peut ajouter que le problème n'est pas uniquement technique ou didactique mais constitue en réalité un véritable enjeu sociétal. La complexité croissante des systèmes numériques est telle qu'elle est depuis plusieurs décennies hors de portée de l'individu. Seuls des collectifs en conservent la maîtrise. Mais ces collectifs sont généralement issus d'entreprises privées et acquièrent ainsi un pouvoir considérable qui les place de plus en plus en position de force face aux organisations gouvernementales. Le mouvement littéraire cyberpunk né dans les années 1960 pour les œuvres les plus précoces, avec des chefs de file comme Philip K. Dick ou William Gibson, a anticipé dans une grande mesure des événements qui sont devenus à présent réalité. Dans « le neuromancien » [3] de William Gibson, publié en 1984, Le héros, Henry Dorsett Case, se voit privé de sa capacité de se connecter au réseau ce qui précipite sa chute. Notre dépendance collective à Internet n'est aujourd'hui probablement pas très différente de celle décrite dans ces ouvrages audacieux, il y a une quarantaine d'années. Par ailleurs, le temps où la complexité combinée à des capacités d'évolution propre placeront les systèmes numériques hors de portée de tout collectif humain, un thème pendant longtemps cher à la science-fiction, commence à émerger dans des scénarios réalistes. Les formations d'ingénieur se doivent d'incorporer aujourd'hui des enseignements de responsabilité sociétale des entreprises qui sont toutefois tournés essentiellement vers le développement durable, la question de l'énergie et le réchauffement climatique. Il y a à la connaissance des auteurs peu de développement autour de la digitalisation

croissante de la société. Le thème de l'éthique des algorithmes se développe néanmoins rapidement en recherche. On peut imaginer que cette problématique devienne rapidement incontournable tant des objets technologiques qui incorporent de l'IA font désormais partie de notre quotidien. Pour ajouter un peu aux perspectives, on se contente ici de citer un article récent [4] qui plaide pour la bonne connaissance des couches profondes dans la bonne utilisation des algorithmes. Quoi qu'il en soit, ces questions pour incongrues qu'elles peuvent sembler, doivent être appréhendées par les organisations formatrices pour qu'enseigner les systèmes numériques en 2050 relève encore d'une problématique démocratique dans une société progressiste.

#### 5. CONCLUSION

Depuis l'apparition du premier microprocesseur à quelques milliers de transistors, les systèmes numériques ont beaucoup évolué au point qu'ils semblent changer de nature épistémologique. Enseigner de manière ascendante, des sciences fondamentales jusqu'aux systèmes du marché actuel dans une démarche quasi-exhaustive est devenu impossible. Il appartient aux établissements de trouver un juste milieu entre des connaissances théoriques des couches profondes et des compétences opérationnelles qui permettent de les mettre en œuvre dans la pratique. La démarche « compétences » formalise dans une bonne mesure le compromis à trouver mais le chantier est important et probablement encore en cours. Par ailleurs, au-delà des aspects techniques, il existe également une problématique de type RSE, responsabilité sociétale des entreprises, qui ne s'est pas encore saisi de ces questions. La littérature, le cinéma ont leur mot à dire dans le cadre des cursus de formation des ingénieurs qui sont avant tout des citoyens soucieux des grandes valeurs humanistes que les programmes des établissements se doivent d'intégrer.

#### Bibliographie

- [1] Mathieu, Henry et Fanet, Hervé. Physique des semi-conducteurs et des composants électroniques : cours et exercices corrigés. Paris: Dunod, 2009.
- [2] Bernard, Jean-Michel, Hugon, Jean et Le Corvec, Robert. De la logique câblée aux microprocesseurs. 2. Applications directes des circuits fondamentaux. 2e éd. revue et corrigée. Paris: Eyrolles, 1979.
- [3] Gibson, William. Neuromancer. New-York, Ace edition, 1984.
- [4] Tsamados, A., Aggarwal, N., Cowls, J. et al. The ethics of algorithms: key problems and solutions. *AI & Soc* (2021). <https://doi.org/10.1007/s00146-021-01154-8>
- [5] Etzioni, Amitai and Etzioni, Oren. Incorporating Ethics into Artificial Intelligence. *The Journal of Ethics*, vol. 21, num. 4, Dec. 2017. <https://doi.org/10.1007/s10892-017-9252-2>