



HAL
open science

Dis maman (ou papa), c'est quoi un algorithme dans ce monde numérique ?

Aurélien Alvarez, Thierry Viéville

► **To cite this version:**

Aurélien Alvarez, Thierry Viéville. Dis maman (ou papa), c'est quoi un algorithme dans ce monde numérique ? . Images des Mathématiques, 2014. hal-00926333

HAL Id: hal-00926333

<https://inria.hal.science/hal-00926333v1>

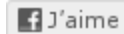
Submitted on 3 Mar 2014

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.

5 janvier 2014

1 commentaire — commenter cet article



Objet du mois

Dis maman (ou papa), c'est quoi un algorithme dans ce monde numérique ?

Aurélien Alvarez et Thierry Viéville

La presse nous en parle tout le temps désormais [1]. Nos systèmes numériques en sont gorgés... ce qui nous donne le meilleur itinéraire routier en est un... on fait atterrir un avion en plein brouillard grâce à un autre... les bourses financières les utilisent pour décider combien coûtera le prix du blé ou du riz qui nourrit les enfants du monde, etc. Ce sont... les algorithmes. « Les algues en quoi ??? »

De puis des années, les chercheurs ont pris la peine d'expliquer aux adultes ce concept levier de l'informatique et des sciences du numérique. On obtient un algorithme lorsque l'on a évacué la pensée d'un procédé et réduit les choses à un simple calcul.

« Un algorithme, c'est tout simplement une façon de décrire dans ses moindres détails comment procéder pour faire quelque chose. Il se trouve que beaucoup d'actions mécaniques, toutes probablement, se prêtent bien à une telle décortication. Le but est d'évacuer la pensée du calcul, afin de le rendre exécutable par une machine numérique (comme un ordinateur par exemple). On ne travaille donc qu'avec un reflet numérique du système réel avec qui l'algorithme interagit. »



Philippe Flajolet



Ainsi, **Philippe Flajolet** nous détaille sur **)(interstices** ce qu'est un **algorithme** pendant que **Gérard Berry** en parle dans un un petit film amateur sur **l'histoire des idées qui ont fait l'informatique**, à destination des collégiens. On peut aussi partir d'une **anecdote**, pour faire comprendre que réduire un procédé à un algorithme consiste à exprimer la solution à un problème donné de manière à ce que même un lendemain de très grosse fête, on puisse très facilement l'utiliser : aucune information implicite, aucun élément de contexte ne doit être pré-supposé, tout doit être détaillé, au moment où il y en a

besoin.

Dans un monde devenu numérique il est essentiel de faire comprendre dès le plus jeune âge cette idée fondamentale. Sans cela, elle ou il ne comprendra pas la différence entre l'intelligence mécanique et l'intelligence humaine. Donc ne saura pas se positionner correctement par rapport à ces systèmes numériques. Et alors, très naturellement, ces objets virtuels seront personnifiés, vus comme magiques, donc domineront l'enfant, et non l'inverse. Il est donc urgent de ne plus attendre pour leur apprendre cela [2].

Mais comment expliquer cela à nos enfants ?

En jouant. Par exemple, au « robot-idiot ». Qui doit sortir d'un petit labyrinthe que l'on aura construit dans le séjour en déplaçant quelques tables ou chaises, ou en dessinant à la craie sur le sol de la cour. Celui qui joue le rôle du robot n'a pas le droit de comprendre le langage humain, mais juste un langage très limité pour effectuer une action décomposée en étapes élémentaires.



Comment sortir d'un labyrinthe ?

qui permettra d'avancer plus efficacement. Avec un crayon à papier et une gomme, on pourra mémoriser la valeur et l'effacer ensuite.

Le jeu pourra se compliquer s'il y a une porte (concrétisée par un objet quelconque) qui peut-être fermée ou ouverte, sans qu'on le sache à l'avance. Il faudra alors introduire une condition dans notre algorithme : « si la porte est fermée alors [fais le tour] »... mais bien entendu il faudra expliquer en détail ce que veut dire « fais le tour » ! Pour le robot, il y aura alors deux paquets de cartes à choisir selon la condition.



Gérard Berry

L'enfant qui imitera le robot ne pourra que :

- (i) avancer d'un pas ;
- (ii) tourner à gauche d'un quart de tour ;
- (iii) tourner à droite d'un quart de tour.

On fabriquera des petites cartes à jouer en découpant un vieux carton en petits carrés avec les mots « avancer », « gauche », « droite ». Et on lui donnera une séquence de ces cartes qui sera son « algorithme ». Il devra exécuter cet algorithme sans « réfléchir » (gare au mur - et à la rigolade - s'il y a un bug !). Ensuite on aura sûrement envie de ne pas répéter « avance d'un pas, avance d'un pas » mais « avance de trois pas ». Donc l'instruction aura une valeur variable



Ce qui arrivera en cas de bug !

On fera ensuite un labyrinthe en spirale [3], comme un escargot. Et là on proposera d'utiliser une boucle « tant que tu n'es pas sorti du labyrinthe, avance tout droit jusqu'au prochain carrefour puis prends à droite ». À ce stade, nos cartes à jouer deviennent trop lourdes à utiliser, la consigne sera simplement écrite sur un papier.

Une autre question va surgir chez les enfants les plus grands : mais comment sortir d'un labyrinthe inconnu ? C'est-à-dire de tous les labyrinthes du monde avec le même algorithme ? Est-ce que « avance tout droit puis prends à droite dès que possible » est un bon algorithme pour tous les labyrinthes ? Eh bien non ! Car on risque fort de tourner en rond... D'où peut-être l'idée de semer des petits cailloux sur le chemin ?



Labyrinthe digital situé à l'entrée de la cathédrale de Lucques en Italie

On

pourrait ainsi avancer tout droit en semant des petits cailloux puis toujours prendre le chemin le plus à droite qui n'a pas de petits cailloux ? Donc toujours aller vers ce qui n'a pas encore été exploré ? Essayons et... tiens-tiens... ça a l'air de toujours fonctionner...

Mais pourrions-nous le faire alors sans les petits cailloux ? Justement, il se trouve figurez-vous que l'**algorithme de Pledge**... [stop] ! Il est sûrement l'heure du goûter :-).

Passer du jeu à la séance de cinéma

Allez, il est temps de se reposer et de regarder (en cliquant par exemple sur l'image ci-dessous) un petit **film** de trois minutes [4] :

Voici son contenu :

« Pour qu'un ordinateur fasse un truc tout seul, un truc simple comme une addition, ou un truc compliqué comme piloter un vaisseau, faut lui expliquer dans les moindres détails tout ce qu'il doit faire. Mais les terriens [...] savent programmer des machines pour qu'elles fassent des trucs toutes seules et qu'elles ne se trompent jamais [par rapport à ce qu'on leur a demandé]. »

« Ce qui permet de dire à un ordinateur tout ce qu'il doit faire s'appelle un algorithme. Un algorithme, c'est une façon très efficace de faire les choses sans avoir besoin de réfléchir ! Par exemple quand tu traverses la rue, pour éviter de te faire écraser, tu regardes à gauche puis à droite puis à gauche. C'est très efficace et c'est toujours la même méthode ; c'est ça un algorithme ! Un algorithme, c'est une méthode efficace pour faire faire quelque chose, même à une machine complètement stupide. »



« Celui qui a inventé ça s'appelait **Al-Khwarizmi**. C'était un arabe du IXe siècle. Il avait été chargé par le calife de trouver une méthode infaillible pour résoudre des problèmes comme calculer les impôts ou mesurer les réserves d'eau par exemple. Et c'est lui qui a compris comment envisager toutes les possibilités, tout le temps. Plus tard, on a repris cette méthode et on l'a appliquée aux ordinateurs. »

Vous venez de partager la première leçon d'informatique théorique avec votre enfant de 6-12 ans.

Que venons-nous d'apprendre ensemble ici ?

Quelque chose de parfaitement inutile, mais totalement indispensable à notre éducation au numérique. Ne cherchez pas dans vos anciens manuels scolaires, cette **leçon de choses** n'y est très probablement pas. Tout d'abord vous venez de lui apprendre que pour exprimer un algorithme il y a **quatre ingrédients** : une séquence d'instruction, des variables, des tests et des boucles. Il se trouve qu'avec ces ingrédients nous pouvons (faire) exécuter tous les algorithmes du monde. Donc tout ce qui se passe dans un smartphone, une tablette, une télévision numérique, un robot ou un ordinateur se réduit à ces ingrédients (en quantité géante). Cela signifie que nous avons caractérisé ce qu'est l'intelligence mécanique. Sauf bouleversement de la science, aucun système artificiel, aucun robot, ne saura faire autre chose qu'exécuter ces algorithmes.

Ce jeu du « robot-idiot » n'est pas du tout original, il correspond



Al-Khwarizmi (env. 783, env. 850)

à une **activité débranchée** que l'on appelle la « **tortue-logo** » ; et il semble que ce soit la façon la plus répandue d'apprendre les ingrédients des algorithmes aux enfants. Et ça marche bien ! Tant mieux... nos enfants ne sont pas là pour servir de cobaye à une expérimentation didactique, mais apprendre de manière plaisante ce dont ils ont besoin. Oui, vous venez faire avec lui quelque chose de parfaitement inutile, bien que totalement indispensable à son éducation au numérique :

comprendre avec lui comment fonctionne un ordinateur !

Dites lui bien que les ordinateurs ne font pas plus que ça, car c'est vrai. De manière fabuleusement rapide et avec des algorithmes fabuleusement plus longs, certes. Mais pas plus. Ce que nous disons ici est une forme popularisée de la **thèse de Church-Turing** qui fonde la science informatique. Mais ne l'embêtez pas [encore] avec ça :-).



Alonzo Church (1903–1995)

La rédaction d'Images des maths et les auteurs remercient pour leur relecture attentive, les relecteurs Claire Wenandy, Michel Mouyssinat et Nicolas Bedaride.

Contentons-nous de partager ce jeu du « robot-idiot » puis de lui montrer tranquillement au cours du temps que plein de choses qu'il fait avec son cerveau (compter, trier des objets, rechercher un mot dans un texte, etc.) peuvent être faites par une machine, car cela se réduit à un algorithme. Mais que faire un joli dessin, choisir sa ou son meilleur-e ami-e, inventer une danse, appartiennent à une autre forme d'intelligence, très différente. L'enfant a donc appris en jouant le premier rudiment de pensée algorithmique, et il retiendra les quatre ingrédients (séquence d'instructions, valeur variable, test et boucle). Plus tard, quand il rencontrera ce formalisme, il aura peut-être oublié le jeu du « robot-idiot » ou le film rigolo que nous avons vu. Lui, oui. Mais pas son cerveau. Car c'est à travers des gestes sensori-moteurs que les concepts les plus abstraits se forment, c'est à travers les mises en situation les plus simples que se forme son bel et jeune esprit. Contrairement aux robots et aux ordinateurs qui, eux, restent de parfaits imbéciles.

P.S. :

La rédaction

d'Images des maths et les auteurs remercient pour leur relecture attentive, les relecteurs Claire Wenandy, Michel Mouyssinat et Nicolas Bedaride.



Alan Turing (1912-1954)

Notes

[1] De plus en plus souvent, des algorithmes décident de notre rapport au monde, comme on peut le lire sur ce **blog**.

[2] Rapport de l'Académie des sciences - mai 2013 : **Il est urgent de ne plus attendre. L'enseignement de l'informatique en France.**

[3] Wikipédia nous enseigne que la **cathédrale de Lucques** comporte un des plus petits labyrinthes d'église. Il est gravé sur un des piliers du portique de la façade et mesure environ 50 cm de large. Les fidèles suivaient le parcours du doigt : c'est un labyrinthe **digital**. L'inscription latine gravée sur la droite du

labyrinthe dit : « Hic quem Creticus edit Daedalus est laberinthus de quo nullus vadere quivit qui fuit intus ni Theseus gratis Ariane stamine jutus », ce qu'on peut traduire par : « Ceci est un labyrinthe que bâtit **Dédale le Crétois**, duquel personne, y ayant pénétré, ne put sortir sauf **Thésée**, grâce au fil d'**Ariane** ».

[4] Grâce à **Tralalère**, **Xprod**, **Inria**, avec **Universcience** et la plume d'**Audrey Mikaëlian**, les **Sépas** sont une 20taine de pépites de science attachées à une personne qui a fait avancer la connaissance et racontées aux enfants. Contenu scientifique réalisé grâce à l'aide de **Nicolas Rougier**, **Sylvie Boldo**, et **Joanna Jongwane**

Affiliation des auteurs

Aurélien Alvarez : Université d'Orléans , **Thierry Viéville** : Chercheur Inria, équipe mnemosyne.

Pour citer cet article : **Aurélien Alvarez** et **Thierry Viéville**, « **Dis maman (ou papa), c'est quoi un algorithme dans ce monde numérique ?** » — *Images des Mathématiques*, CNRS, 2014.

En ligne, URL : <http://images.math.cnrs.fr/Dis-maman-ou-papa-c-est-quoi-un.html>

Si vous avez aimé cet article, voici quelques suggestions automatiques qui pourraient vous intéresser :

- **Dis maman (ou papa), comment on cache des secrets dans le monde numérique ?**, par **Aurélien Alvarez** et **Thierry Viéville**
- **Dis papa (ou maman), comment arrivent les bugs dans le monde numérique ?**, par **Aurélien Alvarez** et **Thierry Viéville**
- **Les tables de multiplication**, par **Étienne Ghys**