La Tétralogique.

Prologue.

Les exposés qui suivent présentent les bases, peu enseignées, des fondements algorithmiques des mathématiques constructives et de l'informatique théorique en vue d'une transposition à la physique. La physique classique est analogique : elle s'est développée sur le dogme du "Tout est continu" avec cette idée sous-jacente que le discret n'y est présent que par dépit, parce que nos moyens calculatoires semblent tributaires du discret. Toutefois, nous verrons que ce que nous prenons pour du dépit est, en réalité, une limitation essentielle, connue sous le nom de Thèse de Church-Turing, sans doute aussi impossible à transgresser qu'une loi de la nature. La physique classique aurait pu être digitale, basée sur le dogme contraire du "Tout est discret" et le continu n'y aurait été présent que par commodité, parce que certains calculs sont plus faciles à conduire dans le cadre de l'analyse que dans le cadre de l'arithmétique. La question demeure : pourquoi l'option retenue fut-elle analogique, nécessité ou simple héritage d'un choix historique? Nous verrons que l'arithmétique discrète possède le même pouvoir d'expression universelle que l'analyse infinitésimale et qu'il est possible de défendre cette thèse que c'est le point de vue digital qui est le mieux adapté à une description saine du monde sensible.

Les sujets que nous voulons traiter sont vastes et délicats : ils passent en revue la crise des fondements que les mathématiques ont connue vers 1900, les recommandations d'Hilbert, les découvertes logiques fondamentales faites par Gödel vers 1930, leur élargissement à la théorie de la calculabilité dû à Kleene, Church et Turing quelques années plus tard et enfin le renouveau du constructivisme en sciences. Les rapports étroits avec l'approche digitale de la physique déjà envisagée par Feynman font partie intégrante de cette présentation et, dans un certain sens, la justifient à nos yeux de physicien.

La littérature abondante sur le sujet ne brille pas partout d'une clarté extrême à tel point que le profane est assuré de se perdre. La raison principale de cet état de fait est que cette discipline, passablement complexe en soi, se trouve à l'intersection de la logique, des mathématiques, de la physique et de l'informatique théorique. Selon la corporation à laquelle ils appartiennent et selon qu'ils privilégient l'approche axiomatique ou algorithmique, les auteurs utilisent les mêmes mots dans des sens complètement différents, usant et abusant des doubles et contre emplois. Cette constatation nous a amené à préfacer l'exposé de la théorie par la présentation d'un glossaire qui fixe, une fois pour toutes, le vocabulaire de base tel que nous l'utilisons. Le corps du texte est ensuite scindé en trois parties. La première s'intéresse au $6^{ème}$ problème d'une liste qu'Hilbert a proposé à la sagacité des mathématiciens du $20^{ème}$ siècle naissant et qui en comprend 23. La deuxième partie étudie l'universalité et l'indécidabilité dans le cadre de l'approche algorithmique de Turing et la troisième partie fait de même dans le cadre de l'approche axiomatique de Gödel. Quelques autres problèmes issus de la liste d'Hilbert, les numéros, 1, 2, 3, 8 et 10, surgiront çà et là, balisant notre promenade de façon cohérente.

Les programmes exemplatifs sont volontairement écrits en langage Mathematica. Ce langage est particulièrement adapté à l'illustration simultanée des points de vues axiomatique et algorithmique. Ce n'est pas un hasard si son concepteur, Stephen Wolfram, est également l'auteur de l'ouvrage, "A New Kind of Science", qui nous sert de référence récurrente. Que l'on partage entièrement ou partiellement les conjectures parfois extrêmes qu'il y défend, force est de reconnaître que pour ce qui concerne l'ensemble des résultats établis avec certitude, on ne trouvera nulle part ailleurs un étalage comparable d'érudition présentée avec autant de clarté.

Nous avons volontairement tenté d'expliquer des sujets compliqués dans un langage accessible. Les entorses à la rigueur sont dès lors inévitables et nous avons tenté de les minimiser. Ne pas adopter ce point de vue reviendrait à écrire un ouvrage de plus, superflu pour le spécialiste et illisible pour le profane.

Glossaire commenté des mots techniques en usage courant.

Ensembles.
Une majorité de mathématiciens admet que la théorie des ensembles offre un cadre idéal pour le développement des mathématiques. Bien qu'une minorité agissante dont nous reparlerons conteste ce point de vue, le fait est que tous les concepts mathématiques sont exprimables dans le langage de la théorie des ensembles. On formalise, par exemple, la notion d'entier naturel à partir de l'ensemble vide et des extensions suivantes :

```
Nest[Union[#,{#}]&,{},5]
```

$0=\varnothing,\ 1=\{\varnothing\},\ 2=\{\varnothing,\{\varnothing\}\},\ 3=\{\varnothing,\{\varnothing\},\{\varnothing,\{\varnothing\}\}\},\ \cdots,\ n+1=n\cup\{n\},\ \cdots$

La théorie des ensembles a connu des débuts difficiles. Dans une première version, qualifiée ultérieurement de « naïve », Cantor a proposé de définir l'ensemble comme une "collection" d'objets qui partagent une propriété distinctive. Il était parfaitement conscient que remplacer le mot ensemble par le mot collection ne faisait guère avancer les choses aussi a-t-il précisé que la définition repose sur l'un ou l'autre des deux axiomes complémentaires suivants :

- l'axiome d'extension, $X=Y \Leftrightarrow \forall z(\, z\in X \Leftrightarrow z\in Y\,)$, qui affirme qu'un ensemble est défini par ses éléments et

- l'axiome de compréhension, $\exists X\forall z(\, z\in X \Leftrightarrow P(\,z\,))$, qui affirme que toute propriété, P, donne naissance à un ensemble.

L'ensemble n'est pas la propriété qui le définit, c'est le résultat final qui compte, c'est-à-dire les objets sélectionnés, par exemple, {a, c, b}. L'ordre dans lequel on écrit les éléments n'a pas d'importance et aucune répétition ne fait de différence : {a, b, c} ou {a, a, c, b} désignent le même ensemble. Si l'on veut que l'ordre joue un rôle, on ne parle plus d'ensemble mais de liste (synonyme : suite), par exemple, la liste {1, 2, 3, 4} diffère de la liste {1, 3, 2, 4}.

Les axiomes de Cantor peuvent paraître inoffensifs et cependant ils définissent des ensembles contradictoires tel, « L'ensemble, A, des entiers caractérisables complètement par une phrase rédigée en français correct et comportant moins de 100 caractères prélevés dans l'alphabet {a,b, …, z, _} ». Or l'entier, x, défini par, « x est le plus petit entier non définissable par une phrase française de moins de cent caractères » ne comporte que 96 caractères : c'est une définition parfaitement valable dans le système de Cantor d'un nombre qui devrait simultanément appartenir à A et à son complémentaire, une contradiction manifeste si l'on s'en tient à une logique binaire. Nous verrons le moment venu comment sortir de l'ornière.

Langages.
On note, Λ_K, l'ensemble infini, des mots finis écrits dans l'alphabet fini, $\{a_1, a_2, \ldots, a_K\}$, comportant K caractères. On range les mots de Λ_K dans l'ordre canonique en les classant dans l'ordre des longueurs croissantes puis, subsidiairement, dans l'ordre lexicographique. On appelle langage tout sous-ensemble de Λ_K. Un langage peut être fini, composé des trois mots, {a, ab, ba} ou infini, par exemple l'ensemble des mots, $\{\varepsilon, a, aa, aaa, aaaa, \ldots\}=\{a^n\}$, où ε est le mot vide. On note qu'un paramètre au moins est indispensable pour noter le cas infini. De même, l'ensemble des nombres premiers est assimilable à un langage infini dans l'alphabet de notre choix, binaire, {0, 1} ou décimal, {0, 1, 2, …, 9}.

Problème et Schéma de problèmes.
Un problème est une question porteuse de sens qui appelle une réponse comme dans l'exemple, "Que vaut la somme, (2+2) ?". Un schéma de problèmes exige davantage : en incorporant dans son énoncé un ou plusieurs paramètres, il pose une question générique qui s'adresse à un ensemble de données non explicitement spécifiées. L'addition, "Que vaut la somme (m+n) ?", est un schéma de problèmes dont chaque instance sous-entend des valeurs pour m et n. La lourdeur de l'expression "Schéma de problèmes" fait qu'on ne fait habituellement aucune différence entre "Problème" et "Schéma de problèmes".

Problème de décision.
On nomme ainsi tout problème qui n'appelle que les réponses "oui" ou "non". Exemple : "Soit un ensemble, A, l'élément, a, appartient-il à l'ensemble A?" pose un problème de décision. On pourrait penser que le caractère rudimentaire des réponses possibles limite sérieusement le champ des problèmes de décision mais il n'en est rien. On transforme, par exemple, le problème de l'addition en un problème de décision en ajoutant un paramètre : "A-t-on que m+n=p?". De toute évidence certaines instances sont positives, elles appellent la réponse "oui" et d'autres sont négatives, elles appellent la réponse "non".

La notion de langage formalise la notion de problème de décision "par ses solutions". Plus précisément, il suffit de considérer que les mots qui appartiennent au langage encodent les instances positives de ce problème de décision. Par exemple, le langage des mots, $\{3, 4, 5\}$, encode les instances positives du problème, "n est supérieur à 2 et inférieur à 6". Autre exemple, le langage des mots, $\{0+0=0,\ 0+1=1,\ 1+0=1,\ 1+1=2,\ 1+2=3,\ \cdots\}$, construits sur l'alphabet, $\{0,1,\cdots,9,+,=\}$, regroupe les instances positives du problème de l'addition dans N. Il existe quantité de façons de coder le même problème dans un alphabet différent, par exemple, au travers des mots du langage, $\{a^m b^n c^{m+n}\} = \{abcc,\ abbccc,\ aabccc,\ abbbcccc,\ aabbcccc,\ \cdots\}$, construit sur l'alphabet, $\{a,b,c\}$. Un automate qui est capable de compter les lettres dans l'ordre peut répondre affirmativement ou négativement à la question de l'appartenance d'un mot donné au langage. Nous verrons, sous peu, que la définition d'un problème "par ses solutions" est plus générale que la définition traditionnelle "par l'énoncé de ses données". Insistons sur le fait qu'un problème de décision n'exige qu'une réponse positive ou négative et pas du tout qu'on détaille un mode quelconque de résolution pratique.

Procédure (synonyme : algorithme).
Une procédure est une suite (nécessairement ordonnée) finie d'instructions qui est sensée résoudre toutes les instances d'un problème posé. Chaque instance implique un calcul autonome selon la valeur des données mais tous les calculs suivent le modèle unique de la résolution de principe dictée par l'algorithme. La notion de procédure justifie que l'on s'intéresse aux schémas de problèmes. En effet, tout problème qui ne comporte qu'une seule instance admet toujours l'une ou l'autre des deux procédures triviales suivantes, qui ne calculent rien du tout mais qui se contentent d'afficher "oui" ou "non". C'est pour éviter cette tricherie que l'on ne s'intéresse habituellement qu'aux problèmes multi instances qui exigent un algorithme non trivial qui affiche la réponse correcte dans tous les cas (synonyme : qui décide le schéma de problèmes).

Procédure effective.
Une procédure est effective si elle résout toutes les instances du problème posé en un temps fini. Une procédure effective s'arrête donc toujours en affichant la réponse correcte. La méthode enseignée à l'école primaire pour additionner deux entiers quelconques est une procédure effective. Pour toutes sortes de raisons, banales ou profondes, tous les problèmes ne sont pas solubles par une procédure effective. Par exemple, il n'existe aucune procédure effective capable de trouver le plus petit entier naturel qui n'est pas la somme de quatre carrés parfaits. C'est la conséquence prévisible d'un théorème dû à Laplace selon lequel tout entier peut s'écrire comme somme de quatre carrés. Il en résulte que toute tentative procédurale est vouée à l'échec : elle tourne sans jamais s'arrêter pour la simple raison qu'elle cherche quelque chose qui n'existe certainement pas.

Voici deux exemples qui semblent similaires et qui cependant posent autant de questions irrésolues à ce jour. Existe-t-il une procédure effective capable de trouver le plus petit entier pair (≥4) qui n'est pas la somme de deux nombres premiers? Aucune démonstration n'est connue que ce nombre n'existe pas : si on lance un programme informatique à sa recherche, celui-ci semble ne jamais s'arrêter sans que l'on sache avec certitude si oui ou non il finirait par s'arrêter un jour. Le problème de Collatz est du même tonneau. Pour rappel, ce problème cherche un entier, $u_0 > 1$, qui engendre une suite infinie au travers de la récurrence suivante qui démarre à, $n = 0$,

$$si \;\; u_n = 1 \;\; alors \;\; stop$$
$$si \;\; u_n \;\; pair \;\; alors \;\; u_{n+1} = u_n / 2$$
$$si \;\; u_n \;\; impair > 1 \;\; alors \;\; u_{n+1} = (3u_n + 1)/2$$

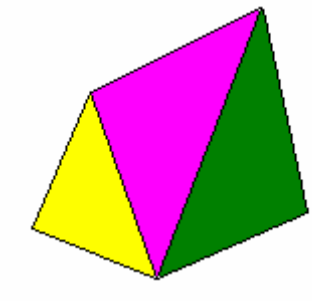
Aussi loin qu'on poursuive les recherches, il semble que toutes les suites finissent par retomber à 1. L'entier, $u_0 = 27$, est le premier, dans l'ordre naturel, qui engendre une suite assez longue mais le fait est qu'elle demeure finie :

```
{27,41,62,31,47,71,107,161,242,121,182,91,137,206,103,155,233,350,175,263,395,593,890,445,668,
334,167,251,377,566,283,425,638,319,479,719,1079,1619,2429,3644,1822,911,1367,2051,3077,4616,
2308,1154,577,866,433,650,325,488,244,122,61,92,46,23,35,53,80,40,20,10,5,8,4,2,1}
```

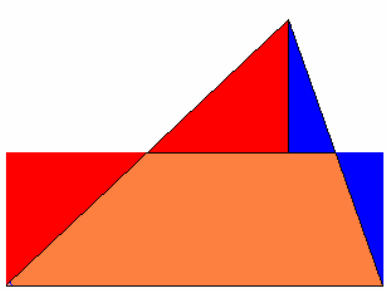
Existence d'une procédure effective : l'exemple du troisième problème de Hilbert.
Le 3ème problème de Hilbert est un exemple de problème de décision soluble par une procédure effective. Il considère deux polyèdres quelconques mais de même volume et il pose à leur sujet la question suivante : peut-on par une procédure effective décider si oui ou non il est possible de scier le premier selon des plans et de recoller les morceaux, en nombres finis, pour recomposer le second ?

En fait, le troisième problème de Hilbert généralise, à trois dimensions, le problème bidimensionnel de Wallace, Bolyai et Gerwien, à savoir : étant donnés deux polygones de même aire, existe-t-il une procédure effective qui décide s'il est possible de découper le premier par un nombre fini de coups de ciseaux rectilignes et de réarranger les morceaux pour recomposer le second ? Ce problème est visiblement équivalent à celui de la quadrature d'un polygone donné. Cette quadrature est effectivement toujours possible, autrement dit le problème bidimensionnel est décidable. Il existe, de fait, une procédure effective qui répond systématiquement "oui" à la question posée et qui va même plus loin en détaillant, en quatre

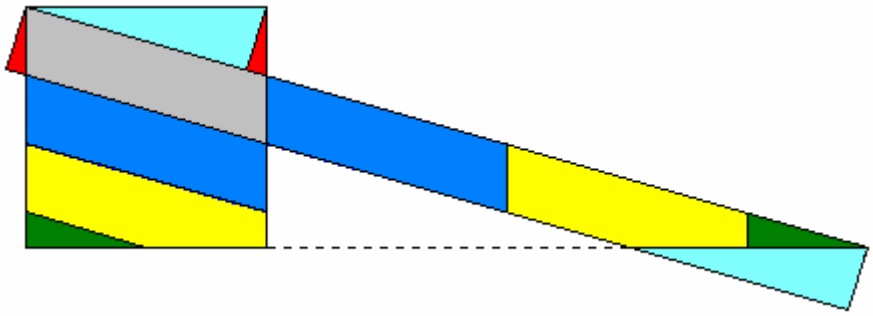
temps, comment il faut procéder (ce que le problème de décision ne demande pas mais qui peut le plus peut évidemment le moins !).



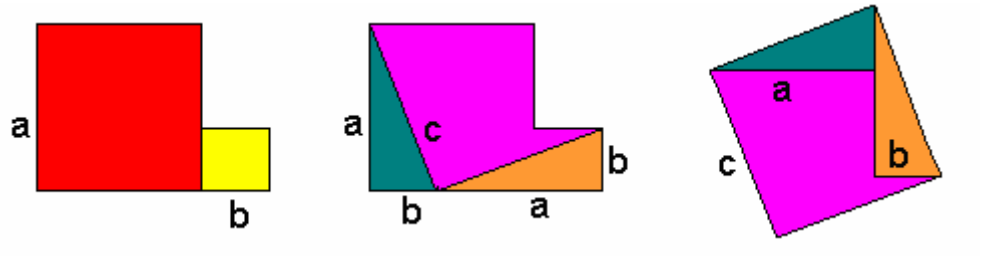
1. Triangulation du polygone donné.



2. Découpage de chaque triangle en un rectangle équivalent.



3. Quadrature de tout rectangle. (Le découpage exige de connaître préalablement le côté, c, du carré équivalent mais la construction est classique : dans tout triangle rectangle la hauteur relative à l'hypoténuse est moyenne proportionnelle entre les segments qu'elle y détermine, $h^2=ab$)



4. Découpage et reconstruction d'un carré équivalent à deux carrés donnés (Pythagore).

Dans la liste des 23 problèmes posés par Hilbert, le troisième a été résolu le premier : dès 1902, Dehn démontrait que le problème tridimensionnel est effectivement décidable et que, contrairement au cas bidimensionnel, il mène le plus souvent à une réponses négative. En

fait, l'opération de recomposition du deuxième polyèdre à partir du premier n'est possible que si et seulement si les polyèdres de même volume ont le même invariant de Dehn.

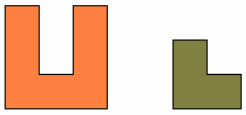
L'invariant de Dehn, $I = \sum_{arêtes} \ell \otimes (\delta + Q\pi)$, se calcule sous la forme d'un produit tensoriel dans $R \otimes R/\pi Z$ où ℓ représente la longueur de chaque arête et δ est l'angle dièdre associé, calculé modulo n'importe quel multiple rationnel de π. L'appellation « indice » de Dehn conviendrait aussi bien mais « invariant » se réfère au fait que sa valeur n'est affectée par aucun découpage, en vertu de la double linéarité, $\ell \otimes (\alpha + \beta) = \ell \otimes \alpha + \ell \otimes \beta \qquad (\ell + m) \otimes \alpha = \ell \otimes \alpha + m \otimes \alpha$.
Deux polyèdres quelconques de même volume ont généralement un invariant différent d'où la réponse négative au troisième problème de Hilbert.

Par exemple, l'invariant de Dehn d'un cube est nul car, $\ell \otimes \pi/2 = 0$, mais celui d'un tétraèdre quelconque ne l'est pas : la cubature de ce dernier est donc impossible sauf exceptions liées à des mensurations très particulières qui annuleraient I.

Inexistence d'une procédure effective : l'exemple du pavage du plan.
Il existe des problèmes de décision très simples qui ne sont pas décidables et le pavage du plan est un exemple célèbre. Il s'énonce comme suit : étant donné un ensemble illimité de tuiles planes dont les formes polygonales sont quelconques mais respectent obligatoirement un des N gabarits imposés, il n'existe aucune procédure effective qui décide dans tous les cas s'il est possible de paver le plan sans recouvrements ni trous.

Il faut bien comprendre que le problème posé est générique : il exige une procédure immuable qui apporte une réponse quelles que soient ses instances particulières. Il est tout à fait possible que certaines instances reçoivent une réponse aisée par le biais d'un raisonnement plus ou moins astucieux.

- Par exemple, aucun pavage n'est manifestement possible avec les gabarits suivants, c'est évident pour l'intuition et il ne devrait pas être difficile d'en formaliser une preuve :

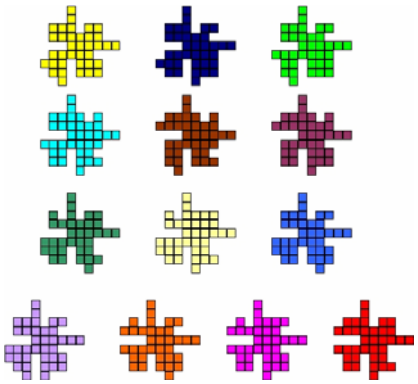


Gabarits imposés : pavage du plan impossible.

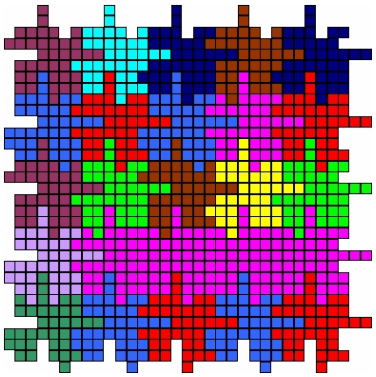
- Par contre si les gabarits imposés sont des rectangles de tailles différentes, disons trois pour fixer les idées, la réponse est évidemment positive : il suffit d'aligner les rectangles identiques selon des lignes jointives et parallèles ce qui peut d'ailleurs se faire d'une infinité de manières.

Gabarits imposés Pavage du plan possible

- D’autres instances beaucoup plus compliquées du même problème peuvent parfaitement recevoir une réponse positive : qui croirait a priori que le pavage suivant (non périodique !), emprunté à O. Bodini, soit possible à partir de 13 gabarits ? Les premiers pavages non périodiques ont été découverts par Penrose.



Gabarits imposés



Pavage du plan

Toutefois aucun problème de décision ne s'embarrasse des cas particuliers, ce qu’il exige c’est une réponse dans tous les cas à l’aide d’un algorithme unique. Dans le cas du pavage du plan, le premier résultat incontestable a été obtenu par Berger qui a démontré qu’aucune procédure effective ne pouvait exister dans le cas du pavage par des polyominos. Pour rappel, les polyominos sont des assemblages connexes de n carrés identiques, les dominos correspondant au cas particulier, n=2.

Implémentation d'une procédure effective.

Le fait qu'il existe une procédure de décision effective relative à un schéma de problèmes n'implique nullement que celle-ci soit facile à mettre en œuvre ou même connue. Voici, pour la petite histoire, l'exemple fameux de la différence de statut entre les problèmes posés par la dérivation ou l'intégration des fonctions élémentaires. Appelons élémentaire toute fonction qui applique de façon répétée un nombre quelconque mais fini de fonctions algébriques, exponentielles ou logarithmiques à une forme rationnelle de la variable, x. Les problèmes posés demandent s'il existe une procédure effective qui décide si une fonction élémentaire donnée possède une dérivée ou une primitive elle-même élémentaire et, le cas échéant, si la réponse est positive, qui la calcule. La réponse au problème de dérivation est connue et enseignée à l'école secondaire : une procédure existe effectivement qui répond toujours "oui" et qui va même plus loin en détaillant un calcul pratique de la dérivée qui s'applique à tous les cas. Il est moins connu qu'une procédure de décision identique (qui répond toujours "oui") existe également pour le problème de la primitivation. Dû à Risch, l'algorithme du calcul effectif de la primitive est largement implémenté dans les logiciels modernes de calcul formel. Seuls certains cas résistent eu égard à la difficulté pratique, mais non théorique, qu'il y a à le programmer complètement. Voici l'exemple spectaculaire du calcul de la primitive d'une fonction élémentaire, que le lecteur peut essayer sur son logiciel d'intégration formelle :

$$\int \frac{(3x+1)\ell n(x)+3x^2+x+(x^2+x+1)\sqrt{x+\ell n(x)}}{(x\ell n(x)+x^2)\sqrt{x+\ell n(x)}+x^2\ell n(x)+x^3}dx \cong 2\ell n(x+\sqrt{x+\ell n(x)})+2\sqrt{x+\ell n(x)}$$

Ensembles finis et infinis.
Un ensemble est infini s'il peut être mis en bijection avec l'une de ses parties sinon il est fini. Par exemple, l'ensemble des entiers naturels est infini car on a la bijection,

$$\{1,\ 2,\ 3,\ 4,\cdots\}$$
$$\Updownarrow$$
$$\{2,\ 4,\ 6,\ 8,\cdots\}$$

Cette prouesse est, de fait, impossible à réaliser avec un ensemble fini. Un ensemble fini, A, qui possède n éléments est dit de cardinal, n. Deux ensembles finis ont même cardinal s'il existe une bijection entre leurs éléments respectifs. L'ensemble, P(A), des sous-ensembles de A est également fini et son cardinal vaut exactement 2^n. Voici l'exemple, n=3 :

$$A=\{a,b,c\} \qquad P(A)=\{\emptyset,\{a\},\{b\},\{c\},\{a,b\},\{a,c\},\{b,c\},\{a,b,c\}\}$$

On peut vouloir considérer l'ensemble, P(P(A)) des sous-ensembles de P(A) et même poursuivre la manœuvre autant de fois que l'on veut : les ensembles résultants voient leur cardinal exploser tout en restant fini.

La théorie des ensembles serait simple si l'on s'en tenait aux ensembles finis qui ne conduisent à aucun paradoxe connu. Toutefois, Cantor a souhaité aller beaucoup plus loin et assurer un statut légal aux ensembles infinis. Il a franchi le pas audacieux selon lequel l'existence d'une bijection garantit l'égalité des cardinaux des ensembles même lorsqu'ils sont infinis. Il a introduit la notation $\aleph_0$ pour le cardinal de N et par voie de conséquence pour tous les ensembles, dits dénombrables, qui peuvent être mis en bijection avec lui. Un ensemble infini est donc dénombrable si ses éléments peuvent être mis en bijection avec ceux de l'ensemble des entiers naturels, N={ 1, 2, 3, …}. On ne s'étonnera pas de ce que {2, 4, 6, 8, 10, 12, …} et {1, 2, 3, 4, 5, 6, …} puisse être de même cardinal puisque c'est le propre des ensembles infinis de pouvoir être mis en bijection avec une de leurs parties.

Plusieurs mathématiciens contemporains de Cantor, à commencer par l'illustre Poincaré, ont désapprouvé cette démarche : ils estimaient que jouer avec une notion aussi peu intuitive que l'infini présentait des risques pour la cohérence des mathématiques. Certes, même pour Poincaré, il n'est pas question de se passer de l'infini « potentiellement en devenir » telle que l'implique la notion de limite. Si Poincaré protestait, c'était pour mettre en garde contre les ambitions de Cantor qui allaient bien au-delà en prétendant manipuler l'infini « actuel », en particulier en le soumettant à une véritable arithmétique transfinie. Cependant les vues de Cantor eurent le bonheur de recevoir très vite l'appui enthousiaste de Hilbert qui encouragea la communauté des mathématiciens à profiter de ce qu'il appela « Le Paradis de Cantor » et, de fait, elles finirent par s'imposer au terme d'une période probatoire qui montra qu'aucunes des prédictions funestes de Poincaré ne semblaient se réaliser.

Tout sous-ensemble d'un ensemble dénombrable est soit fini soit dénombrable. Autrement dit, il n'existe pas d'ensemble infini "intermédiaire" dont le cardinal serait inférieur à $\aleph_0$.

Ensembles récursivement énumérables.
Un ensemble, X, est récursivement énumérable (synonyme : listable) s'il existe une procédure effective capable d'énumérer tous les éléments de l'ensemble dans une liste. Aucun ordre particulier n'est imposé et les répétitions sont admises, certains auteurs tolérant même les lacunes. La seule exigence est que la liste soit complète dans l'inventaire des éléments de X. L'énumération peut théoriquement se faire d'au moins n! manières distinctes si l'ensemble est de cardinal fini, n, et d'une infinité de manières sinon.

Par exemple, la liste, x_n = {0, 0, *, *, 2, 2, *, *, 4, 4, …}, est recevable comme énumération de l'ensemble des entiers pairs. Elle dicte les éléments dans un ordre prévisible et il importe peu qu'elle présente des lacunes et des répétitions. Evidemment on peut faire plus simple, par exemple, {0, 2, 4, 6, …}.

On peut noter, x_n, la liste résultant de l'énumération mais on peut tout aussi bien remplacer cette liste, x_n, par la fonction, x[n] = x_n. Il y a autant de fonctions que de listes qui conviennent, et la présence de lacunes dans la liste correspond au fait que la fonction, x(n), n'est que partiellement définie.

Tout ensemble infini récursivement énumérable est évidemment dénombrable. Voici quelques exemples d'énumérations effectives d'ensembles. Ils ont en commun de proposer une bijection avec N que l'on reconnaîtra à l'existence simultanée des fonctions, x[n] et n[x]. Rappelons que cela est un luxe puisqu'une surjection dans N suffirait.

1) L'ensemble des entiers pairs positifs est dénombrable et récursivement énumérable :

x = {2, 4, 6, 8, 10, 12, …}
n = 1 2 3 4 5 6 …

$$x[n\_] := 2n$$
$$n[x\_] := \frac{n}{2}$$

Exemple : x[2]=4 et n[4]=2.

2) L'ensemble des entiers relatifs est dénombrable et récursivement énumérable :

x = {0, 1, -1, 2, -2, 3, -3, 4, -4, 5, -5, …}
n = 1 2 3 4 5 6 7 8 9 10 11 …

$$x[n\_] := (-1)^{n+1}\,\text{Floor}\left[\frac{n+1}{2}\right]$$
$$n[x\_] := 2\,\text{Abs}[x] - \text{If}[x > 0, 1, 0]$$

Exemple : x[5]=-2 et n[-2]=5.

3) L'ensemble des entiers premiers est dénombrable et récursivement énumérable :

p = {2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, …}
n = 1 2 3 4 5 6 7 8 9 10 11, …

Dans ce cas, la numérotation par une formule compacte n'est pas connue et il faut recourir à un algorithme qui n'est qu'une généralisation procédurale de la notion de formule. Il suffit de tester la primalité des entiers dans l'ordre naturel pour obtenir la numérotation désirée.

4) L'ensemble des couples d'entiers est dénombrable et récursivement énumérable :

{x, y} = {(1, 1), (2, 1), (1, 2), (3, 1), (2, 2), (1, 3), (4, 1), (3, 2), (2, 3), …}
n = 1 2 3 4 5 6 7 8 9 …

```
n[{x_, y_}] := 1/2 ((x + y - 2)^2 + x + 3 y - 2)
{x[n_], y[n_]} := {1/2 (3 Floor[(-1 + Sqrt[8 n - 7])/2] + Floor[(-1 + Sqrt[8 n - 7])/2]^2 - 2 n + 4), 1/2 (-Floor[(-1 + Sqrt[8 n - 7])/2] - Floor[(-1 + Sqrt[8 n - 7])/2]^2 + 2 n)}
```

Exemples : n[2,1]=2 et {x[2],y[2]}=(2,1).

Les couples, (x, y) ont été ordonnés prioritairement par valeurs de (x+y) croissantes et subsidiairement par valeurs de x décroissantes au sein des couples. Les triplets, les quadruplets, etc, subiraient un sort comparable avec les complications d'écritures qu'on devine pour les codages.

5) L'ensemble, Λ_K, des mots de longueurs finies écrits dans un alphabet fini, $\{a_1, a_2, \ldots, a_K\}$, comportant K caractères, est dénombrable et récursivement énumérable. Par exemple, voici les classements de Λ_2 et de Λ_3 dans l'ordre canonique :

Λ_2 = {0, 1, 00, 01, 10, 11, 000, 001, 010, 011, 100, 101, 110, 111, 0000, …},
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 …

Λ_3 = {0, 1, 2, 00, 01, 02, 10, 11, 12, 20, 21, 22, 000, 001, 002, 010, 011, 012, 020, …},
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 …

L'ensemble, Λ_K, est effectivement énumérable par une procédure effective qui s'écrit :

```
code[K_, liste_] := (K^Length[liste] - 1)/(K - 1) + liste.Table[K^j, {j, Length[liste] - 1, 0, -1}]
liste[K_, n_] := PadLeft[IntegerDigits[Floor[n + (1 - K^Floor[Log[K, n K - n + 1]])/(K - 1)], K],
                 Floor[Log[K, n K - n + 1]]]
```

Exemples : code[3,{0,2,0}]=19 et liste[3,19]= {0,2,0}.

L'ensemble des chaînes finies de caractères écrites dans un alphabet fini est dénombrable et récursivement énumérable par la même procédure. Une procédure étant toujours, par définition, une chaîne de caractères de longueur finie, on en conclut que l'ensemble des procédures est dénombrable car c'est un sous-ensemble infini de Λ_K.

6) L'ensemble des sous-ensembles *finis* de N ou, plus généralement de n'importe quel ensemble dénombrable est dénombrable et récursivement énumérable. Il suffit de coder chaque sous-ensemble par l'entier dont l'écriture binaire inversée ne comporte des '1' qu'aux positions qu'il indice. Par exemple, le sous-ensemble, {2,4} est codé par l'entier, 1010_2=10.

```
codesubset[liste_] := Apply[Plus, 2^(liste - 1)]

decode[code_] := Flatten[Position[Reverse[IntegerDigits[code, 2]], 1]]
```

Voici le début du classement valable pour N :

x = {{1},{2},{1, 2},{3},{1, 3},{2, 3},{1, 2, 3},{4},{1, 4},{2, 4}, …}
n = 1 2 3 4 5 6 7 8 9 10 …

Ensembles non dénombrables.
Tous les ensembles infinis ne sont pas dénombrables. Dans l'exemple 6 précédent, la finitude des sous-ensembles est essentielle. La situation change radicalement si on considère tous les sous-ensembles de N, *finis ou infinis* (synonyme : parties). On démontre par l'absurde que l'ensemble, P(N), des parties de N ou d'ailleurs de n'importe quel ensemble dénombrable n'est pas dénombrable. La réduction suivante par l'absurde est classique.

Notons, $A=\{a_1,a_2,a_3,\cdots\}$, l'ensemble dénombrable de départ et, $S=\{s_1,s_2,s_3,\cdots\}$, l'ensemble de ses sous-ensembles, supposé lui aussi dénombrable. Il suffit de considérer l'ensemble, $D=\{a_i | a_i \notin s_i\}$, pour voir que D ne peut figurer nulle part dans la liste des éléments de S dont il fait pourtant légitimement partie. On déduit de cette absurdité que S ne peut pas être dénombrable.

A partir de là, on voit la possibilité d'engendrer des ensembles de plus en plus vastes : on part de N (ou de n'importe quel ensemble dénombrable) et en considère l'ensemble de ses parties puis, itérativement, l'ensemble des parties de l'ensemble obtenu à l'étape précédente. On obtient alors une suite de cardinaux dits transfinis :

SCT : $\aleph_0 < 2^{\aleph_0} < 2^{2^{\aleph_0}} < \cdots$

La notation exponentielle est pure convention qui repose sur l'analogie suivante : l'ensemble des sous-ensembles d'un ensemble fini comportant n éléments en comporte exactement 2^n. Avec ces notations, on a les résultats fondamentaux suivants :

1) L'ensemble des sous-ensembles (finis ou infinis) d'un ensemble dénombrable, en particulier l'ensemble des langages sur un alphabet donné, n'est pas dénombrable mais de cardinal, $2^{\aleph_0}$.

2) L'ensemble des réels dans l'intervalle]0,1[(ou d'ailleurs dans n'importe quel intervalle multidimensionnel fini ou infini) n'est pas dénombrable. On note, C, le cardinal de ce continuum. Le développement décimal illimité de tout réel présent dans l'intervalle]0,1[pouvant être mis en bijection avec une partie de N, on a que, $C = 2^{\aleph_0}$.

3) L'ensemble des suites finies ou infinies d'entiers est aussi de cardinal, $2^{\aleph_0}$.

4) L'ensemble des fonctions de N dans N est de cardinal, $2^{\aleph_0}$. *Par contre, l'ensemble des fonctions de R dans R est de cardinal,* $2^{2^{\aleph_0}}$.

La non dénombrabilité de l'ensemble des réels présents dans l'intervalle]0,1[peut aussi être établie par une technique de diagonalisation plus une réduction par l'absurde. Si les réels dans l'intervalle]0,1[pouvaient être numérotés sans omission, on pourrait écrire :

1 0.**5**22165211…
2 0.0**5**6842352…
3 0.25**8**708855…
4 0.874**1**30456…
5 0.0006**5**7011…
…

Tout réel dont le n^{ième} chiffre après le point décimal différerait, pour tout n, de celui noté en gras qui occupe le n^{ième} emplacement diagonal, serait nécessairement absent du tableau qui s'avèrerait donc largement incomplet contrairement à l'hypothèse de départ.

Le lecteur qui veut vraiment comprendre l'argument diagonal a intérêt à trouver la faille dans le raisonnement suivant. Dressons un tableau similaire avec tous les rationnels compris entre 0 et 1 rangés dans l'ordre, {1/2, 1/3, 1/4, 2/3, 1/5, 2/4, 1/6, 2/5, 3/4, 1/7, …}, facile à débrouiller. Cela donnerait :

1/2 = 0.**5**000000…
1/3 = 0.3**3**33333…
1/4 = 0.25**0**0000…
2/3 = 0.666**6**666…
1/5 = 0.2000**0**00…
…

Cette liste est complète : elle contient tous les rationnels compris entre 0 et 1. De plus, on sait que le développement décimal de chacun est périodique sur le long terme. En appliquant l'argument diagonal, il est très facile de construire beaucoup de nombres qui ne figurent nulle part dans cette liste : il suffit que leur k^{ième} chiffre diffère du k^{ième} chiffre diagonal dans le tableau. Cela laisse une grande latitude de choix et on pourrait être tenté de penser qu'avec un peu de méthode, il devrait être possible de s'arranger pour construire un développement périodique à partir d'un certain rang. Cela serait catastrophique cependant puisque cela signifierait qu'on aurait réussi à construire un rationnel qui ne fait pas partie de la liste réputée complète des rationnels !

Par ailleurs, on évitera de confondre les notions de cardinal et d'ordinal. Le cardinal d'un ensemble est unique et il indique combien l'ensemble possède d'éléments. La notion d'ordinal, dont nous ferons peu d'usage, est nettement moins intuitive. Elle concerne les ensembles bien ordonnés. Un ensemble est totalement ordonnable si on peut trouver une procédure réflexive, transitive et antisymétrique qui décide, pour tout couple d'éléments, lequel "précède" l'autre. Un bon ordre exige en plus que tous ses sous-ensembles non vides (y compris lui-même) possèdent un plus petit élément. L'ensemble, Z, des entiers relatifs possède un ordre total mais pas un bon ordre. Par contre, l'ensemble, N, des entiers naturels est muni d'un bon ordre, l'ordre naturel avec 0 comme plus petit élément. Dans cet ensemble bien ordonné, on construit la suite des ordinaux récursivement comme suit : le premier ordinal est le sous- ensemble singleton qui ne contient que le plus petit élément, 0, et l'ordinal d'ordre, n+1, s'obtient à partir de l'ordinal d'ordre, n, en lui adjoignant le plus petit élément non encore sélectionné. Concrètement, cela donne la suite, $\{1=\{0\},2=\{0,1\},3=\{0,1,2\},\cdots\}$, où chaque ordinal apparaît sans ambiguïté comme l'ensemble de ses prédécesseurs.

Avec cette définition, on voit que tout ordinal possède un seul cardinal. Cependant l'inverse n'est pas obligatoirement vrai d'où la conclusion que la notion d'ordinal est plus fondamentalement riche que celle de cardinal. Si l'ensemble bien ordonné de départ est fini, il n'y a qu'une manière, à un isomorphisme près, de classer ses éléments de sorte que les deux notions se recouvrent exactement. Par contre, si l'ensemble de départ est infini, il peut posséder plusieurs bons ordres non isomorphes auxquels correspondent des ordinaux distincts, alors que rappelons-le, son cardinale est unique.

En suivant la définition, le premier ordinal infini est défini comme l'ensemble des entiers qui le précèdent, on le note, ω, $\omega=\{0,\ 1,\ 2,\ 3,\ \cdots\}$. On peut ensuite définir la suite d'ordinaux :

$$\omega+1=\{0,\ 1,\ 2,\ 3,\ \cdots,\ 0\},\ \omega+2=\{0,\ 1,\ 2,\ 3,\ \cdots,\ 0,\ 1\},\ \ldots,\ \omega+3,\ \ldots$$

jusqu'à tomber sur $\omega + \omega$, qu'on pourrait tout aussi bien noter, $\omega\ 2$, et ainsi de suite, $\omega\ 3$, $\omega\ 4$, …, $\omega\ \ \omega=\omega^2=\{\omega m+n : m,n\in N\}$), ω^3, …, ω^ω, ω^{ω^ω}, etc …, jusqu'à tendre vers la limite d'une tour infinie d'ω empilés. Une étrange arithmétique ordinale transfinie non commutative se met en place, que nous ne détaillons pas. Tous ces ordinaux sont dénombrables : ils possèdent donc tous le même cardinal, $\aleph_0$. Le plus petit ordinal non dénombrable, noté ω_1, est l'ensemble des ordinaux dénombrables. Son cardinal se note, $\aleph_1$. De même, on note ω_2 le plus petit ordinal dont le cardinal, $\aleph_2$, est supérieur à $\aleph_1$ et plus généralement, on note, $\aleph_\alpha$, le cardinal de l'ordinal, ω_α. On a, par définition, que la suite des cardinaux transfinis,

$\aleph_0 < \aleph_1 < \aleph_2 < \cdots$,

est complète ce que n'est pas forcément la suite, STC, $\aleph_0 < 2^{\aleph_0} < 2^{2^{\aleph_0}} < \cdots$.

On a que, $\aleph_0 < 2^{\aleph_0}$, tandis que, par définition, $\aleph_1 \leq 2^{\aleph_0}$. Cantor a conjecturé que, dans cette dernière relation, c'est l'égalité, $\aleph_1 = 2^{\aleph_0}$, qui prévaut. Autrement dit, il a cherché à prouver qu'il n'existe pas d'infini intermédiaire entre $\aleph_0$ et $2^{\aleph_0}$ et, plus généralement, entre deux quelconques des membres de la suite STC des cardinaux transfinis. Cette conjecture porte, dans la littérature, le nom d'hypothèse du continu et elle occupe la première place dans la liste de Hilbert. Le malheureux Cantor s'est, dit-on abîmé la santé en tentant de prouver cette conjecture et le fait est qu'il aurait pu chercher longtemps : Cohen a démontré que cette proposition n'est ni prouvable ni réfutable dans le cadre de la théorie des ensembles, elle y est indécidable. On veut dire par là qu'on pourrait ajouter aux axiomes de la théorie des ensembles n'importe laquelle des variantes suivantes, α = 1, 2, 3, …, de l'axiome, $C = 2^{\aleph_0} = \aleph_\alpha$, sans provoquer d'incohérence.

L'application classique des ordinaux transfinis est de permettre d'étendre le principe de récurrence au-delà des entiers naturels au sein d'une théorie des ensembles munis d'un bon ordre. Nous verrons que c'est par une récurrence transfinie que Gentzen a pu prouver la cohérence des axiomes de l'arithmétique dans le cadre de la théorie des ensembles.

Problèmes (non) énonçables.

Nous avons vu que tout langage définit un problème "par ses solutions". Traditionnellement, on s'attend à pouvoir définir un problème "par son énoncé". Par définition, un énoncé raisonnable est un texte de longueur finie qui fixe les conditions du problème sans ambiguïté. Il n'est pas difficile de voir que la définition par les solutions est plus générale que celle par les énoncés. C'est la conséquence du fait qu'il existe une infinité non dénombrable de langages donc de jeux de solutions, alors qu'il n'existe qu'une infinité dénombrable d'énoncés. Autrement dit, on peut définir "par ses solutions" une infinité non dénombrable de problèmes qui ne peuvent recevoir aucun énoncé fini. A la réflexion, cela n'est pas vraiment surprenant : une infinité non dénombrable de langages sont aléatoires du fait qu'ils regroupent une infinité de mots sans rapports entre eux. Il ne faut pas s'attendre à ce qu'on puisse rassembler ces mots dans un énoncé qui fasse mieux que les incorporer in extenso : un tel énoncé serait fatalement infini et donc déraisonnable.

Sous-ensembles décidables (synonyme : récursifs).

Un sous-ensemble, A, d'un ensemble donné, X, est décidable dans X s'il existe une procédure effective qui est capable de recevoir un élément de X en entrée et, après analyse, de répondre par oui ou par non à la question de l'appartenance de cet élément au sous-ensemble A. Tout langage étant un sous-ensemble de l'ensemble, Λ_K, des mots finis que l'on peut écrire dans un alphabet comportant K lettres, on parle également de langage décidable avec la même signification.

Les sous-ensembles finis sont toujours décidables puisqu'il suffit de comparer tout élément donné à chaque élément du sous-ensemble et la procédure est assurée de se terminer. L'argument ne vaut plus obligatoirement si le sous-ensemble est infini. En effet, un simple argument de comptage permet de se rendre compte que l'immense majorité des sous-ensembles ne peuvent pas être décidables : l'ensemble des sous-ensembles d'un ensemble infini n'est pas dénombrable alors que l'ensemble des procédures effectives l'est. On en conclut qu'il n'existe pas assez de procédures effectives pour décider tous les sous-ensembles d'un ensemble infini. Paradoxalement, nous verrons ultérieurement qu'il n'est pas du tout facile d'exhiber un exemple concret de sous-ensemble non décidable.

Sous-ensembles semi décidables (synonymes : semi récursifs, reconnaissables).
Un sous-ensemble, A, de X n'est que semi-décidable s'il existe une procédure effective qui est capable de recevoir un élément de X en entrée et, après analyse, de répondre par oui à la question de l'appartenance de cet élément à l'ensemble, A, lorsqu'elle a lieu et de répondre non *ou de ne pas répondre* dans le cas contraire. Nous verrons que la raison de l'absence éventuelle de réponse n'est pas imputable à un dérangement de l'automate qui fait tourner la procédure, à une erreur de programmation ou à n'importe quelle bêtise de ce genre : elle est liée à un bouclage interne impossible à contourner.

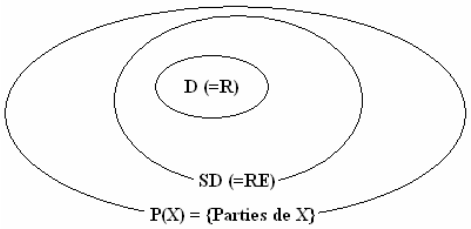
Semi décidable = récursivement énumérable.
Il se fait que les notions de semi décidabilité et d'énumérabilité récursive se recouvrent exactement. L'équivalence entre ces deux notions s'établit comme suit. Un sous-ensemble récursivement énumérable est certainement semi décidable. De fait, on attend d'une procédure de semi-décision qu'elle réponde affirmativement lorsque l'élément fait partie du sous-ensemble. Il suffit de lancer la procédure d'énumération et on est certain de rencontrer tout élément qui est dans ce cas. Il est plus compliqué d'établir qu'un sous-ensemble semi décidable est récursivement énumérable. En effet, il ne suffit pas de considérer les éléments dans l'ordre des longueurs croissantes puis, subsidiairement dans l'ordre lexicographique, et de leur appliquer la procédure de semi-décision car cette procédure risque de boucler dès qu'on tombe sans le savoir sur un mot qui n'appartient pas au sous-ensemble. Une astuce est nécessaire et la voici. On considère les éléments dans l'ordre canonique précité, w_1, w_2, w_3, …, et on leur applique la procédure de semi décision pas à pas, une instruction élémentaire à la fois. L'ordre, *{élément, pas}*, dans lequel on procède est important, on le détaille comme suit :

$$(w_1,1)\ (w_1,2)\ (w_2,1)\ (w_3,1)\ (w_2,2)\ (w_1,3)\ (w_1,4)\ (w_2,3)\ (w_3,2)\ \cdots$$

Le fait que l'automate travaille pas à pas empêche toute forme de bouclage. Il ne reste plus qu'à attendre qu'il reconnaisse les éléments qui appartiennent au sous-ensemble, ce qui ne peut manquer de se produire puisque par hypothèse le sous-ensemble est semi décidable. Ceux-ci sont dès lors énumérés sans faute ni omission quoique dans un ordre impossible à prédire.

Décidabilité et semi décidabilité des sous-ensembles d'un ensemble donné.
Soit X un ensemble dénombrable, par exemple l'ensemble des mots sur un alphabet donné et, P(X), l'ensemble de ses parties (= des langages dans l'exemple). Notons, D, (certains auteurs notent R pour récursif) l'ensemble des sous-ensembles de X décidables dans X et notons SD (ou RE pour récursivement énumérable) l'ensemble des sous-ensembles semi décidables dans X. On a évidemment, $D \subseteq SD \subseteq P(X)$, (ou $R \subseteq RE \subseteq P(X)$).

On établit sans peine que l'union et l'intersection de deux sous-ensembles décidables sont décidables. De même, l'union et l'intersection de deux sous-ensembles semi décidables sont semi décidables. On a aussi que le complémentaire d'un sous-ensemble décidable est décidable donc que le complémentaire d'un sous-ensemble semi décidable et non décidable ne peut pas être décidable. Toutefois, lorsqu'un sous-ensemble et son complémentaire sont semi décidables, alors ils sont automatiquement simultanément décidables.



Ces propositions permettent de mieux comprendre pour quelle raison un sous-ensemble, A, de X qui est semi décidable n'est pas forcément décidable. En fait, il ne l'est que si le sous-ensemble complémentaire, X-A, est lui aussi semi décidable. Lorsque c'est le cas, on peut, en effet, lancer en parallèle les deux procédures de reconnaissance, celle qui reconnaît les éléments de A et celle qui reconnaît les éléments de X-A : on est alors certain d'obtenir une réponse pour tout élément. Cette manœuvre ne fonctionne plus si, X-A, n'est pas semi décidable : il arrivera que l'automate ne cesse de calculer sans que l'on puisse faire la différence entre un vrai bouclage et une procédure qui s'éternise mais qui finirait par s'arrêter. Clairement c'est l'incertitude qui plane sur l'arrêt des procédures (des programmes si l'on préfère) qui est à l'origine de cette distinction de classes. Voici, sans justification à ce stade, l'inventaire des propriétés de quelques langages emblématiques en théorie de la calculabilité. Dans cet inventaire, le terme "procédure" peut indifféremment être remplacé par "programme" ou, anticipant quelque peu, plus formellement par "machine de Turing (=MT)" :

$$H = \{\{Proc, w\} : Proc \quad s'arrête \quad sur \quad w\} \qquad H \in RE - R$$

Traduction : il n'existe aucune procédure capable de décider le sous-ensemble des couples, {programmes, données}, pour lesquels il y a arrêt. Une procédure de semi décision existe cependant qui consiste simplement à faire tourner le programme.

$$\overline{L_0} = \{ w : (w = w_i) \wedge (Proc_i \quad semi \quad décide \quad w_i \} \qquad \overline{L_0} \in RE - R$$

Traduction : il n'existe aucune procédure capable de décider le sous-ensemble des mots reconnus par le programme qui porte le même numéro dans la numérotation canonique. Une procédure de semi décision existe cependant.

$$L_0 = \{ w : (w = w_i) \wedge (Proc_i \quad ne \quad semi \quad décide \quad pas \quad w_i \} \qquad L_0 \notin RE$$

Traduction : il n'existe aucune procédure capable de semi décider (a fortiori de décider) le sous-ensemble des mots non reconnus par le programme qui porte le même numéro dans la numérotation canonique.

Tous les sous-ensembles d'un ensemble donné ne sont pas semi décidables donc encore moins décidables. Il se fait que les langages semi décidables sont exactement ceux que l'on peut engendrer à partir d'une grammaire.

Grammaires.

Considérons un langage, L, et posons-nous la question suivante : est-il possible d'énumérer tous ses mots au moyen d'un ensemble de règles grammaticales communes auxquelles les mots du langage obéissent ? Par définition, une grammaire comprend deux alphabets : l'alphabet du langage proprement dit (où on convient de proscrire les majuscules) et l'alphabet des symboles de formation (composé exclusivement de lettres majuscules parmi lesquelles figure obligatoirement le symbole de départ, S). Les règles de production sont toutes du style, $\alpha X\beta \rightarrow w\gamma$, où w désigne n'importe quel mot écrit sur l'alphabet du langage et où les lettres grecques désignent des mots quelconques écrits sur les deux alphabets mis en commun. En particulier, ε désigne le mot vide. Selon le degré de complexité des règles autorisées, on distingue quatre classes de langages formant la hiérarchie, dite de Chomski :

1) Les langages réguliers, générés par une grammaire régulière (R) : $A \rightarrow w; A \rightarrow wB$.
Le langage régulier, $\{ a^n \} = \{ \varepsilon, a, aa, aaa, aaaa, \cdots \}$, est engendré par la grammaire régulière, $\{ S \rightarrow aS, S \rightarrow \varepsilon \}$

2) Les langages générés par une grammaire hors-contexte (HC) : $A \rightarrow \beta$. Le langage, $\{ a^n b^n \} = \{ \varepsilon, ab, aabb, aaabbb, \cdots \}$, est hors contexte mais pas régulier. Il est engendré par la grammaire hors-contexte, $\{ S \rightarrow aSb, S \rightarrow \varepsilon \}$

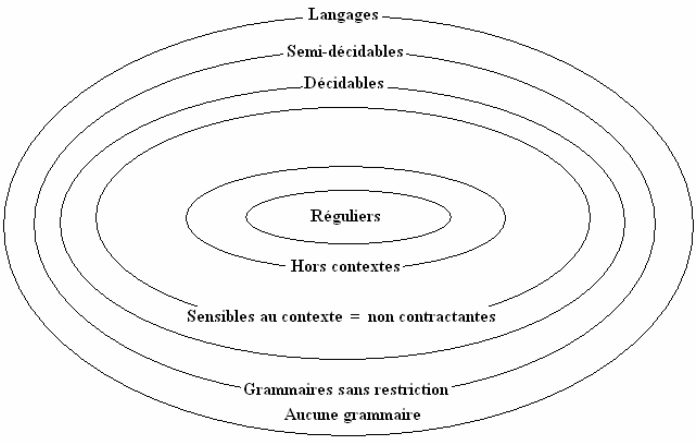
3) Les langages générés par une grammaire sensible au contexte (SC) : $\alpha \rightarrow \beta \qquad (|\alpha| \leq |\beta|)$
Le langage, $\{ a^n b^n c^n \} = \{ \varepsilon, abc, aabbcc, aaabbbccc, \cdots \}$, est sensible au contexte mais il n'est pas hors contexte. Il est engendré par la grammaire hors-contexte,
$\{ S \rightarrow aSBC, S \rightarrow aBC, CB \rightarrow BC, aB \rightarrow ab, bB \rightarrow bb, bC \rightarrow bc, cC \rightarrow cc, S \rightarrow \varepsilon \}$

4) Les langages générés par une grammaire sans restriction, $\alpha \rightarrow \beta$, c'est-à-dire dont certaines règles peuvent être contractantes, $(|\beta| < |\alpha|)$.

Commentaires.
Toutes les grammaires non contractantes mènent à des langages décidables parce que l'application répétée des règles de production ne raccourcit jamais la longueur des mots. Une procédure de décision est alors facile à mettre en place : il suffit d'appliquer systématiquement les règles grammaticales et de générer tous les mots du langage jusqu'à dépasser la longueur du mot donné. On a alors la certitude de pouvoir décider si ce mot appartient au langage.

Rien n'exclut qu'une grammaire contractante engendre un langage décidable, comme dans l'exemple suivant : le langage, $\{ a^n, a^n b \} = \{ \varepsilon, b, a, ab, aa, aab, aaa, aaab, \cdots \}$, est décidable bien qu'engendré par la grammaire contractante, sensible au contexte, $\{ S \rightarrow aS, aS \rightarrow b, S \rightarrow \varepsilon \}$. Cependant, la plupart du temps, les grammaires contractantes mènent à des langages indécidables car a procédure de décision précédente ne s'applique plus du fait que la longueur des mots engendrés par la grammaire peut décroître à tout moment.

Les langages semi décidables sont exactement ceux qui sont engendrés par une grammaire sans restriction et ceux qui ne sont même pas semi décidables ne sont engendrés par aucune grammaire. On a les inclusions suivantes :



Le gros contingent des langages qui ne sont pas semi décidables est fourni par l'ensemble des langages aléatoires qu'aucune grammaire ne structure. Bien qu'ils soient indénombrablement nombreux il est impossible d'en expliciter un seul précisément parce qu'ils sont aléatoires ! Toutefois, et c'est là que se situe le cœur de l'informatique théorique, il existe des langages, tel L_0, qui ne sont pas semi décidables et qui sont pourtant structurés par une définition (non grammaticale d'après ce qui vient d'être dit) qui leur ôte tout caractère aléatoire.

Universalités.

Certains systèmes indécidables jouissent d'une propriété particulièrement intéressante : ils sont universels aux sens de Turing ou de Gödel selon le contexte dans lequel on travaille. Il est prématuré de détailler la signification de ces termes à ce stade de l'exposé. Par contre, il peut être utile de préciser ce qu'ils ne signifient pas. Le vocable universel est, en effet, utilisé dans beaucoup de contextes différents dont le seul point commun est le suivant : il s'agit d'exprimer qu'un système est capable d'effectuer seul un ensemble de tâches qui requièrent habituellement l'intervention d'un ensemble de systèmes. Voici trois exemples.

- Nand ou Nor sont des connecteurs logiques universels au sens de Boole. On veut dire par là que toute formule de la logique propositionnelle faisant intervenir des combinaisons arbitraires de connecteurs, Not, And, Or, Xor, Implies, Equal, …, peut être réécrite en terme du seul connecteur, Nand ou Nor. Ils sont les seuls à posséder cette propriété. Par exemple, on a :

$$And[a,b] = Nand[Nand[a,b],Nand[a,b]]$$
$$Or[a,b] = Nand[Nand[a,a],Nand[b,b]]$$

- Certains réels, infiniment nombreux, sont universels au sens de Borel. On veut dire par là qu'on trouve quelque part dans la succession de leurs décimales n'importe quel groupement de chiffres fixé d'avance y compris celui qui encode l'intégrale de l'œuvre de Shakespeare.

- Il existe des équations différentielles universelles. Ce sont des équations qui jouissent de cette propriété étonnante de posséder des solutions qui approximent, avec un degré de précision arbitrairement grand, n'importe quelle fonction continue donnée d'avance sur n'importe quel intervalle réel. Telle est l'équation de Duffin,

$$ny'''' y'^2 + (2-3n) y''' y'' y' + 2(n-1) y''^3 = 0 \qquad (n > 3, y \in C^n).$$

Automates.
Lorsqu'on sait qu'une procédure effective existe, il encore plus intéressant de connaître un moyen de l'architecturer, sur le papier d'abord puis physiquement ensuite. On appelle automate le résultat concret de cette implémentation. Plus un langage est riche plus l'automate qui le (semi-)décide est sophistiqué.

Les automates finis déterministes (AFD) conviennent pour les langages réguliers.
Les automates à pile déterministe (APD) conviennent pour les langages hors contexte.
Les automates à borne linéaire (ABL) conviennent pour les langages sensibles au contexte.

Tous ces automates sont étudiés dans les cours spécialisés mais, qui peut le plus peut le moins, nous nous concentrerons, le moment venu, sur l'automate qui les émule tous, à savoir la machine de Turing (MT). Nous verrons que la MT semi décide les sous-ensembles semi décidables et la MT qui s'arrête décide les sous-ensembles décidables.

Anticipant à nouveau, signalons qu'un automate à borne linéaire est une machine de Turing qui pour tout mot, w de longueur, $|w|$, écrit sur son ruban d'entrée, s'impose, moyennant le recours éventuel à une extension finie de son alphabet, de ne jamais déborder de cet espace mémoire lors de son exécution. De façon équivalente, sans extension d'alphabet, elle ne consomme pas plus de $k_1|w|+k_2$ cases sur sa bande de lecture-écriture, où k1 et k2 sont des constantes caractéristiques de l'automate. Il ne faut pas chercher ailleurs la raison de la décidabilité des langages sensibles au contexte.

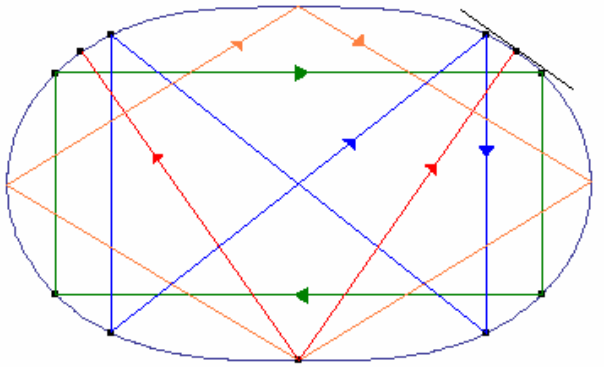
Constructivisme.
La procédure effective est au centre de l'attitude constructiviste. Cette notion n'est pas nouvelle : l'exigence des anciens grecs de se restreindre aux constructions géométriques faisables par la règle et le compas participait de cette idée mais on pouvait n'y voir qu'un caprice ludique. Ce qui est nouveau, c'est que le constructivisme puisse être revendiqué comme une nécessité impérieuse par tous ceux qui envisagent la description des systèmes du monde sensible en privilégiant les grandeurs constructibles et les règles d'évolution programmables par des procédures effectives. Ce qui est surtout nouveau, c'est que cette notion impose des limites qui sont au cœur des révolutions mathématique, informatique et osons-le, physique.

Le constructivisme est une attitude scientifique qui considère qu'un problème n'est résolu que lorsqu'il a reçu une réponse effective. En cela il s'oppose au formalisme qui se satisfait, dans certains cas, d'apprendre que tel problème possède zéro, une ou plusieurs solutions même s'il est incapable de voir à quoi ces solutions ressemblent et tout autant de les calculer avec une précision arbitrairement grande fixée d'avance. Ici le mot problème est pris dans un sens très large comme le montrent les quelques exemples qui suivent.

Premier exemple : une preuve constructive.
La philosophie constructiviste s'applique à la notion de démonstration. Voici une démonstration classique, non constructive, du théorème selon lequel il existe des couples d'irrationnels, a et b, tels que a^b est rationnel. Posons, $a=b=\sqrt{2}$, et formons l'expression demandée, $a^b = \sqrt{2}^{\sqrt{2}}$. De deux choses l'une (Tiers exclu!), a^b est rationnel et le théorème est prouvé ou il est irrationnel et dans ce cas, $(a^b)^{\sqrt{2}}=2$ est rationnel et il répond à la question posée. Dans les deux cas le théorème est démontré. On voit bien que cette démonstration repose entièrement sur le principe logique du Tiers exclu. On ne s'étonnera donc pas d'apprendre que l'école constructiviste refuse de recourir à ce principe dans un cadre non

constructif et qu'elle exige une autre démonstration qui fixe le statut de rationalité de $\sqrt{2}^{\sqrt{2}}$. Pour la petite histoire, $\sqrt{2}^{\sqrt{2}}$ est effectivement irrationnel mais la démonstration constructive est bien moins évidente car elle sollicite davantage l'intuition. C'est d'ailleurs une règle générale que l'attaque constructive d'un problème est plus exigeante donc plus difficile que l'attaque purement formelle.

Deuxième exemple : trajectoires périodiques dans un billard convexe.
On appelle billard convexe une table horizontale dont le bord est une courbe fermée, différentiable deux fois, qui reste constamment d'un même côté de la tangente en chacun de ses points. Une boule de billard, assimilée à un point, circule sans frottement sur cette table décrivant, à vitesse constante, des segments de droite qui ne sont brisés que lors des collisions avec la bande périmétrique. A chaque rencontre avec cette bande, la boule rebondit symétriquement par rapport à la normale au point atteint. Ce problème trouve une transposition immédiate en optique : il suffit de remplacer le bord du billard par un miroir et la boule par un faisceau laser.



Une question classique, qui concerne ce type de billard, est de savoir s'il existe une ou plusieurs trajectoires périodiques, que la boule recommencerait indéfiniment après les avoir parcourues une fois. La réponse formaliste à ce problème est aisée : il existe effectivement une infinité de trajectoires rigoureusement périodiques. Certaines, les plus simples, sont rectilignes : ce sont donc de simples va-et-vient. D'autres sont triangulaires, quadrilatères, etc. Il est, par exemple, très facile de se convaincre de l'existence d'au moins un va-et-vient périodique : la corde de longueur maximale, dont personne ne peut contester l'existence même sans faire le moindre calcul, répond, en effet, à la question posée. Cette corde est nécessairement perpendiculaire aux tangentes au billard aux points qu'elle joint, on dit que c'est une binormale, car si ce n'était pas le cas, elle ne serait certainement pas de longueur maximale : il suffirait, en effet, de déplacer une de ses extrémités et on verrait sa longueur s'accroître. Evidemment si la corde est perpendiculaire à la tangente au billard à chaque extrémité, la réflexion ne peut se faire qu'en revenant sur ses pas et le va-et-vient périodique s'en suit. En fait, on peut montrer qu'il existe toujours une deuxième binormale qui répond également à la question posée.

On constate que cet argument formaliste est purement existentialiste et qu'il ne fournit pas le moyen de construire effectivement ces cordes de longueurs extrêmes. Il se contente de convaincre que ces cordes existent à coup sûr, ce que personne ne conteste. Un mathématicien constructiviste exige davantage : il veut qu'on lui indique une procédure

effective qui est capable de trouver cette corde quel que soit la forme initiale du billard convexe.

Un théorème général dû à Birkhoff promet *au moins* $\phi(n)$ trajectoires n-périodiques (pour n>2) à un billard continu convexe, où ϕ désigne la fonction totient d'Euler (nombre d'entiers positifs inférieurs à n et premiers avec lui, 1 inclus). Par exemple, il existe au moins $\phi(3)=2$ trajectoires triangulaires périodiques dont l'une n'est autre que le triangle inscrit de périmètre maximum. Il est sans doute moins évident de comprendre pourquoi le polygone convexe inscrit, de périmètre maximum, fournit un exemple de trajectoire périodique et encore moins pourquoi il en existe au moins une deuxième mais poursuivre dans cette voie nous écarterait du sujet. La figure ci-dessus montrquelques trajectoires périodiques quadrilatères dans un billard dont l'enveloppe obéit à l'équation polaire, $\rho = \frac{a + cos^2\theta}{a + sin^2\theta}$.

Troisième exemple : les nombres constructibles.
Un mathématicien formaliste ne se pose pas de questions concernant l'existence des nombres. Un irrationnel, un transcendant, un complexe existent un point c'est tout, leur définition suffisant à justifier leur existence. Un constructiviste, par contre, exige qu'on lui indique une procédure effective capable de recevoir un nombre quelconque en entrée et de répondre par oui ou par non à la question : ce nombre est-il irrationnel, transcendant, complexe ?

Un nombre est constructible s'il existe une procédure effective qui explicite ces chiffres dans une base convenue d'avance, le cas échéant en égrenant au compte-gouttes ses décimales s'il s'avère qu'elles sont en nombre infini. Les entiers sont constructibles et les rationnels le sont également eu égard à leur développement décimal périodique. Mais que faut-il penser des irrationnels ?

En imposant de ne construire les figures qu'avec la règle et le compas, la géométrie d'Euclide fournit un élément de réponse à cette question : elle explicite une procédure effective qui est capable de construire certains irrationnels particuliers dont l'écriture ne fait intervenir que des rationnels subissant, autant de fois que l'on veut et dans un ordre quelconque, les opérations, +, -, *, / et √, par exemple, $\alpha = (1+2\sqrt{2/3})/(2+\sqrt{3})$. Les irrationnels « constructibles par la règle et le compas » sont tous racines d'un polynôme à coefficients rationnels de degré irréductible, 2^n. Dans l'exemple cité, on aurait de fait, $25+120\alpha-174\alpha^2-72\alpha^3+9\alpha^4=0$. Les impossibilités que l'on associe généralement aux problèmes fameux, quadrature du cercle, duplication du cube et trisection de l'angle se réfèrent toutes au fait qu'aucun des nombres suivants, $\sqrt{\pi}$ et $\sqrt[3]{2}$, et, pour un angle a quelconque, la racine réelle de l'équation, $4x^3-3x=a$, n'est racine d'une équation irréductible de rang 2^n.

Toutefois, ce n'est pas parce qu'un nombre est non constructible avec la règle et le compas qu'il ne l'est pas avec un outillage plus perfectionné. Il est, de fait, parfaitement possible de trisecter n'importe quel angle en recourrant à une courbe auxiliaire appropriée, une hyperbole, par exemple. Ce que cela signifie, c'est que la règle et le compas représentent ensemble une procédure effective particulière dont le pouvoir de calcul reste très limité. L'exposé consacré aux machines de Turing montrera comment repousser les limites de la calculabilité jusqu'à un mur infranchissable.

Tous les réels ne sont certainement pas constructibles, loin de là, puisque leur infinité est non dénombrable alors que l'ensemble des procédures effectives l'est. De toute évidence il n'y a pas assez de procédures effectives pour construire tous les réels. En formalisant l'étude des

réels, la théorie des ensembles est donc clairement non constructive et en travaillant sur le corps des réels, la physique théorique moderne emprunte le même chemin. Nous verrons que ce n'est ni une fatalité ni même une nécessité : on trouve des physiciens et non des moindres, tel Feynman, qui pensent que c'est imprudent et qu'un retour au constructivisme est non seulement possible mais souhaitable.

Les mathématiciens formalistes ont toujours pensé que les constructivistes se compliquaient bien inutilement la vie avec leurs scrupules algorithmiques. L'attaque constructive de n'importe quel problème est toujours plus exigeante donc plus complexe et il ne faut pas chercher ailleurs les raisons de la prédilection des mathématiciens pour le point de vue formaliste. Toutefois les informaticiens ont montré que tout calcul est constructiviste par essence. Dès lors la question se pose de convaincre les physiciens qu'une description raisonnable du monde sensible gagnerait peut-être également à respecter ce point de vue.