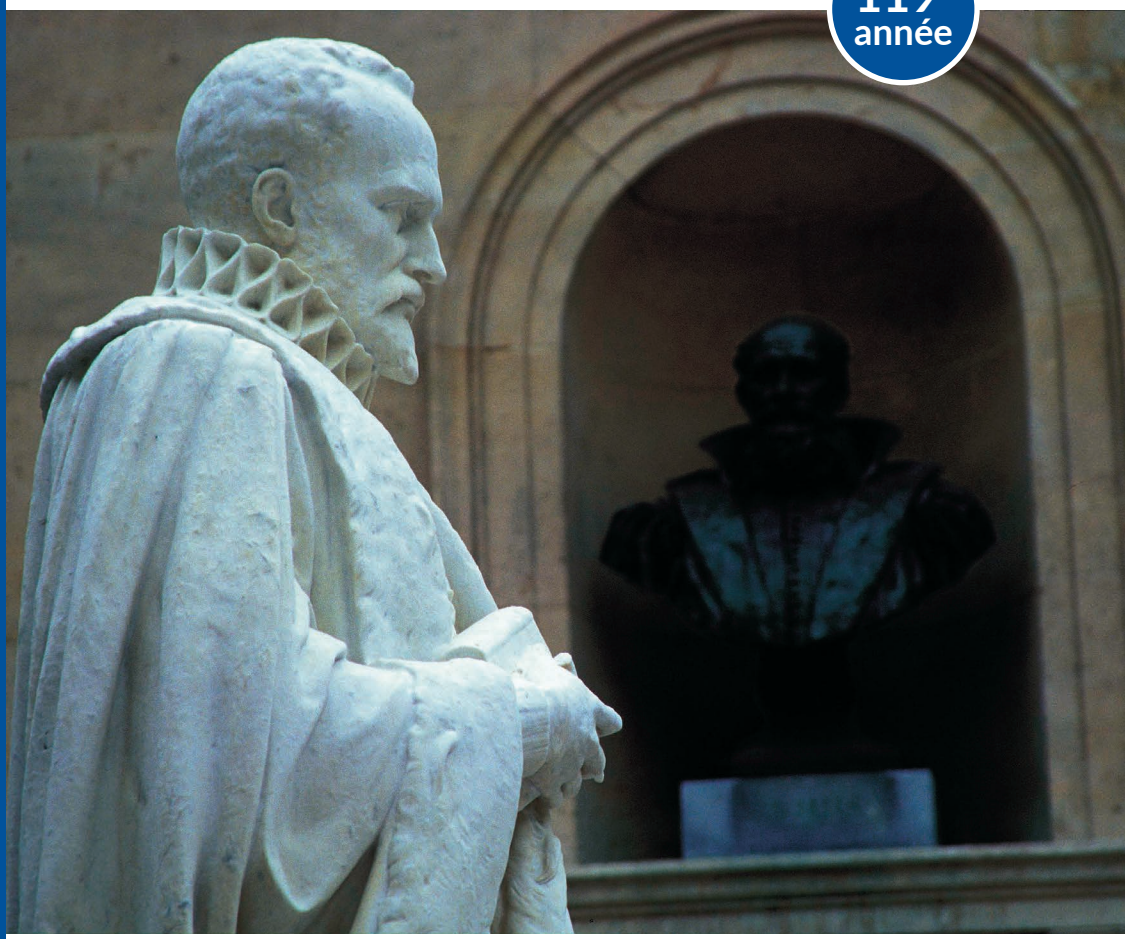
ANNUAIRE du **COLLÈGE DE FRANCE** 2018 - 2019

Résumé des cours et travaux

119e année



COLLÈGE DE FRANCE
1530

INFORMATIQUE ET SCIENCES NUMÉRIQUES*
(CHAIRE ANNUELLE 2018-2019)

Rachid GUERRAOUI

Professeur à l'EPFL de Lausanne,
professeur invité au Collège de France

Mots-clés : algorithme, algorithmique, calculabilité, universalité, complexité

La leçon inaugurale « L'algorithmique répartie : à la recherche de l'universalité perdue » prononcée le 25 octobre 2018 (https://www.college-de-france.fr/site/rachid-guerraoui/inaugural-lecture-2018-2019.htm) ainsi que la série de cours et séminaires « L'algorithmique répartie » sont disponibles, en audio et en vidéo, sur le site internet du Collège de France (https://www.college-de-france.fr/site/rachid-guerraoui/course-2018-2019.htm), ainsi que le colloque « Taking stock of distributed computing » (https://www.college-de-france.fr/site/rachid-guerraoui/symposium-2018-2019.htm).

L'ALGORITHMIQUE RÉPARTIE

CONTEXTE

Les fondamentaux théoriques du numérique supposent pour la plupart qu'un algorithme s'exécute sur une machine de Turing : une seule machine à la fois. Cependant, la grande majorité des algorithmes aujourd'hui s'exécutent, chacun, sur plusieurs machines de Turing en même temps. Les algorithmes modernes sont, autrement dit, fondamentalement répartis. Il y a plusieurs raisons à cela.

Dans le cas extrême de l'« infiniment grand » comme pour *blockchain* par exemple, l'algorithme (sous-jacent à la monnaie virtuelle *bitcoin*) s'exécute sur un

* Chaire créée en partenariat avec l'Inria.

R. GUERRAOUI, « Informatique et sciences numériques », *Annuaire du Collège de France 2018-2019. Résumé des cours et travaux*, 119e année, Paris, Collège de France, 2022, p. 57-63, https://doi.org/10.4000/annuaire-cdf.16784.

très grand ensemble d'ordinateurs géographiquement distants. La taille de cet ensemble de machines de Turing n'est pas déterminée à l'avance et change au cours du temps. Le but de la répartition dans ce contexte est de relier des utilisateurs à travers leurs ordinateurs et le réseau sous-jacent afin de permettre des transactions entre eux, sans passer par un serveur central. Dans le cas d'autres applications pair à pair, le but peut être de diffuser de la musique par exemple, avec comme objectif d'éviter un ordinateur central.

Même lorsqu'un algorithme s'exécute sur une seule machine, aujourd'hui, il s'exécute typiquement sur plusieurs processeurs. Sachant que la vitesse de chaque processeur ne peut quasiment plus s'améliorer du fait de limitations physiques, le seul moyen d'améliorer les performances des architectures matérielles est de mettre plusieurs processeurs sur une même puce et de paralléliser le calcul. C'est l'autre cas extrême de l'« infiniment petit » dans lequel l'algorithme est aussi réparti, mais cette fois sur un tout petit espace dans lequel de nombreuses machines de Turing (processeurs) sont disposés.

Entre les deux cas extrêmes, correspondant dans le premier au besoin de relier plusieurs utilisateurs à travers leurs ordinateurs et au second cas au besoin d'accélérer les performances de calcul en parallélisant, il existe aussi de nombreux cas où les calculs sont répartis pour des raisons de tolérance aux défaillances. On préfère tout simplement ne pas faire reposer toute une application informatique sur une seule machine, mais on la distribue sur plusieurs en dupliquant les parties cruciales du calcul.

Au-delà de ces considérations technologiques, la nature semble aussi conduire à l'étude des algorithmes répartis. Lorsque l'on essaye d'étudier la synchronisation du clignotement des lucioles, les mouvements coordonnés d'un banc de poissons et d'une nuée d'oiseaux, ou encore le comportement collaboratif d'un réseau de neurones, on retrouve des algorithmes répartis.

Problématique

Les théories classiques de calculabilité et de complexité informatiques, inventées pour le numérique séquentiel et centralisé (une seule machine de Turing à la fois), ne peuvent pas s'appliquer au cas de l'algorithmique répartie. La raison est très simple. Alors que dans les théories classiques on suppose que la machine de Turing utilisée fonctionne correctement, émettre une telle hypothèse pour un ensemble de machines est déraisonnable. On souhaite qu'un algorithme qui s'exécute sur un réseau de machines se termine correctement malgré la défaillance d'un sous-ensemble de ces machines. Cette distribution est, comme indiqué ci-dessus, l'une des motivations les plus fréquentes de l'algorithmique répartie. Ce qui peut se calculer sur une seule machine diffère de ce qui peut se calculer sur un ensemble car la notion même de calculabilité change. Le concept d'universalité au sens de Turing change aussi.

En fait, la définition même d'« algorithme correct » doit être revue et corrigée. Dans le cas réparti, cette notion doit prendre en compte l'entrelacement de plusieurs machines de Turing impliquées dans le même calcul, ce qui est loin d'être trivial. Il faut aussi prendre en compte les défaillances et les résultats partiels. La mesure de complexité classique, reflétant en particulier le nombre d'étapes de communication exécutées par un algorithme en fonction de la taille des paramètres d'entrée devient insignifiante. L'important devient le nombre de messages échangés, ou le nombre

d'accès à la mémoire partagée, en fonction du nombre de machines impliquées dans un calcul.

En gros, il existe un décalage entre les théories classiques du numérique, inventées pour le cas séquentiel et centralisé, et les pratiques modernes, intrinsèquement réparties, aussi bien dans ce que l'on appelle « le parallèle » que « le distribué ».

Objectif

La série de cours a présenté les ingrédients d'une théorie de l'algorithmique répartie. Les cours ont couvert les notions d'exactitude d'un algorithme réparti et ont revisité les notions de calculabilité, d'universalité et de complexité dans le contexte réparti. Les cours ont présenté les résultats les plus importants obtenus en algorithmique répartie depuis près d'un demi-siècle et ont souligné les nombreux problèmes encore ouverts. Les cours étaient pour la plupart suivis d'un séminaire spécialisé par un chercheur du domaine considéré comme le spécialiste international. L'année s'est terminée par un colloque réunissant les meilleurs chercheurs du domaine.

Enseignement

Leçon inaugurale – L'algorithmique répartie : à la recherche de l'universalité perdue

L'objectif de ma leçon inaugurale était triple. En premier lieu, il s'est agi d'expliquer que les algorithmes utilisés aujourd'hui sont pour la plupart répartis sur plusieurs machines de Turing. Cela est le cas aussi bien pour les algorithmes déployés à large échelle comme pour le *bitcoin*, que les algorithmes déployés à petite échelle comme à l'intérieur d'un Iphone 8. Cela est aussi le cas pour ce que l'on appelle les « algorithmes naturels » qui sont intrinsèquement répartis. Le deuxième objectif était d'expliquer que cette répartition du numérique crée un décalage entre les fondements théoriques du numérique qui sont basés sur l'idée qu'un programme s'exécute sur une seule machine de Turing à la fois, et les algorithmes modernes qui sont répartis et en utilisent plusieurs en même temps. Il s'est agi ici de motiver le besoin d'une nouvelle théorie. Enfin. Le troisième objectif était de donner quelques exemples de résultats déjà obtenus dans le cadre de cette nouvelle théorie du numérique réparti afin d'intéresser les auditeurs par les cours donnés dans le cadre de cette chaire pendant l'année qui a suivi.

Voici les principaux thèmes abordés pendant la leçon inaugurale :

- « De Mohammed Algorithmi à Alan Turing » ;
- « Connecter des machines de Turing : parallélisme et distribution » ;
- « Naissance d'une discipline : l'algorithmique répartie » ;
- « À la recherche de l'universalité perdue » ;
- « La brève histoire d'un compteur infini » ;
- « Les algorithmes naturels ».

COURS ET SÉMINAIRES

Cours 1 – Qu'est-ce qu'un algorithme réparti correct ? (L'atomicité dans un système réparti)

26 octobre 2018

L'objectif de ce premier cours a été de définir de manière précise ce qu'est un algorithme réparti correct. La propriété de sûreté considérée est que l'on appelle « l'atomicité », ou « la linéarisabilité », définie par référence à la spécification séquentielle d'un objet informatique. La propriété de vivacité considérée est ce que l'on appelle le « wait-freedom » (« absence d'attente »). Les deux propriétés ont été illustrées à travers le théorème d'équivalence entre registres, ainsi qu'un algorithme de mise en œuvre d'un compteur faible.

Voici les principaux thèmes abordés pendant le premier cours :

- « Atomicité » ;
- « Absence d'attente (*wait-freedom*) » ;
- « Théorème d'équivalence entre registres » ;
- « Algorithmique des registres » ;
- « Quelques premiers résultats d'impossibilité ».

Cours 2 – Le pouvoir de la mémoire partagée

9 novembre 2018

Ce cours a présenté comment des processeurs communiquant à travers une mémoire partagée, avec comme seules opérations possibles la lecture et l'écriture, pouvaient résoudre des problèmes de coordination sophistiqués comme la mise en œuvre d'un compteur réparti ainsi que la prise d'un état global atomique. Le cours a permis, à travers ces exemples, d'illustrer les concepts fondamentaux de l'algorithmique répartie, et en particulier les notions d'atomicité et d'absence d'attente.

Voici les principaux thèmes abordés pendant ce second cours :

- « Mise en œuvre d'un compteur faible » ;
- « État global d'un système réparti » ;
- « Exemple de preuve d'atomicité ».

Le cours a été suivi du séminaire de la professeure **Hagit Attiya** (Technion) sur l'algorithmique répartie sans coordination.

Cours 3 – L'impossibilité du consensus

23 novembre 2018

Ce cours a motivé l'importance du fameux problème du consensus, avant d'expliquer en quoi ce concept est central au numérique aujourd'hui. Le cours a ensuite présenté le théorème d'impossibilité du consensus ainsi que sa démonstration dans le cas, relativement simple, de la mémoire partagée. La notion de bivalence a été introduite, tout comme le concept fondamental en algorithmique répartie d'adversaire. Plusieurs corollaires de l'impossibilité du consensus ont ensuite été présentés. Le cours a souligné aussi le lien avec la topologie algébrique pour introduire le séminaire invité sur le consensus ensembliste et la topologie algébrique.

Voici les principaux thèmes abordés pendant ce cours :
- « Mise en œuvre d'un compteur fort » ;
- « Le théorème d'impossibilité du consensus » ;
- « Adversaire, valence et exécution » ;
- « Corollaires de l'impossibilité » ;
- « Consensus ensembliste ».

Le cours a été suivi du séminaire du professeur **Petr Kouznetsov** (Telecom Paris) sur les liens entre l'algorithmique répartie et la topologie algébrique.

Cours 4 – Le théorème d'universalité du consensus

7 décembre 2018

Ce cours a introduit la notion d'universalité en algorithmique répartie en l'illustrant dans le contexte d'une mémoire partagée. Le cours a permis de définir de manière précise ce qu'est un objet informatique universel et a présenté une construction universelle simple fondée sur un objet consensus et des objets registres. Le cours a aussi présenté la généralisation de cette universalité au cas du consensus ensembliste.

Voici les principaux thèmes abordés pendant ce cours :
- « Objet universel » ;
- « Construction universelle » ;
- « Universalité généralisée ».

Le cours a été suivi du séminaire du professeur **Vassos Hadzilacos** (Toronto) sur la hiérarchie du consensus.

Cours 5 – De l'infiniment petit à l'infiniment grand (de la mémoire partagée à l'envoi de messages)

21 décembre 2018

Ce cours a présenté le théorème de transformation de la mémoire partagée en passage de messages. À travers cette transformation, le cours a permis de présenter le concept fondamental de quorum en algorithmique répartie.

Voici les principaux thèmes abordés pendant ce cours :
- « Le théorème d'équivalence entre mémoire partagée et passage de message » ;
- « L'importance des quorums » ;
- « Le théorème des partitions ».

Le cours a été suivi du séminaire du professeur **Hugues Fauconnier** (Paris Diderot) sur la notion de détection de défaillance dans un système réparti.

Cours 6 – Universalité distribuée : le consensus indulgent et l'algorithme Paxos

18 janvier 2019

Ce cours a présenté plusieurs manières de contourner la fameuse impossibilité du consensus. Il a tout d'abord présenté comment le matériel, en particulier dans le cas de la mémoire partagée, permet d'atteindre un consensus réparti. L'hypothèse du partiellement synchrone a ensuite été considérée, ainsi que son impact sur la possibilité du consensus. La mise en œuvre du consensus a permis d'introduire les notions de « leader » ainsi que d'algorithmes « obstruction-free » et « lock-free ».

Voici les principaux thèmes abordés pendant ce cours :
- « Le pouvoir de consensus du matériel informatique » ;
- « Le pouvoir de consensus du temps » ;
- « L'élection d'un leader » ;
- « La puissance de l'aléa pour atteindre le consensus » ;
- « Le réparti comme générateur de nombres aléatoires ».

Le cours a été suivi du séminaire du professeur **Achour Moustéfaoui** (université de Nantes) sur les possibilités d'accord réparti malgré des participants malicieux.

Cours 7 – Les algorithmes dynamiques

15 février 2019

Ce cours a présenté les conséquences de la remise en question d'une hypothèse souvent sous-jacente à l'algorithmique répartie : le nombre de machines impliquées dans le calcul réparti est connu à l'avance. Ce cas statique a été contrasté avec le cas dynamique dans lequel l'ensemble des machines participantes au calcul n'est pas connu à l'avance.

Voici les principaux thèmes abordés pendant ce cours :
- « Le "group membership" » ;
- « Du synchronisme au synchronisme virtuel » ;
- « Une diffusion fiable qui se termine » ;
- « L'importance de l'uniformité ».

Le cours a été suivi du séminaire du professeur **Sébastien Tixeuil** (Paris Sorbonne) sur les algorithmes auto-stabilisants.

Cours 8 – Si *Blockchain* est la solution, quel est le problème ?

1er mars 2019

Ce cours a présenté l'algorithmique sous-jacente à la fameuse cryptomonnaie *bitcoin*. De nombreux parallèles ont été fait entre les algorithmes de consensus présentés dans les cours précédents et la manière avec laquelle *bitcoin* fonctionne. Le cours a ensuite présenté des alternatives frugales au protocole *bitcoin* original.

Voici les principaux thèmes abordés pendant ce cours :
- « *Bitcoin* et *blockchain* » ;
- « L'injustice du minage » ;
- « D'un compteur ordinaire à un compte bancaire » ;
- « Le pouvoir du *gossip* ».

Le cours a été suivi du séminaire de la professeure **Carole Delporte** (Paris Diderot) sur les protocoles de population.

Colloque

Le colloque du 12 avril 2019 a porté sur une théorie unifiée de l'algorithmique répartie.

Voici le programme de ce colloque qui a eu lieu au Collège de France :
- Nir Shavit (MIT) : « High throughput connectomics » ;
- Janna Burman (université Paris Sud) : « Challenges in population protocols » ;

• Yehuda Afek (université de Tel Aviv) : « Distributed computing by rational agents » ;
• Marc Shapiro (Paris Sorbonne) : « Living without consensus » ;
• Valerie King (université de Victoria) : « Byzantine agreement, an old problem in a new world » ;
• M. Potop-Butucaru (Paris Sorbonne) : « Are blockchains a challenge for distributed computing? » ;
• E. Gafni (UCLA) : « *Deus ex machina*: A deus-object: approving genuine imitation of itself » ;
• Sergio Rajsbaum (univsersité de Mexico) : « 25 years of a topological perspective on distributed computing » ;
• A. Bouajjani (Paris-Diderot) : « Systematic design of adequately consistent distributed systems » ;
• P. Fraignaud (Paris Diderot et CNRS) : « Distributed network computing » ;
• Gadi Taubenfeld (Interdisciplinary Center) : « New models for distributed computing » ;
• M. Raynal (IRISA) : « New models for distributed computing » ;
• R. Guerraoui (EPFL et Collège de France) : « Distributed computing: New birth or swan song? ».

Publications

Trois livres liés à la matière du cours

Guerraoui R. et Kouznetsov P., *Algorithms for Concurrent Systems*, Lausanne, EPFL Press, 2018.

Cachin C., Guerraoui R. et Rodrigues L., *Reliable and Secure Distributed Programming*, Berlin/New York, Springer, 2011.

Guerraoui R. et Kapalka M., *Principles of Transactional Memory*, San Rafael, Morgan Claypool, 2010.

Deux livres liés à la chaire

Suite à mon cours au Collège, j'ai aussi été contacté pour écrire deux autres ouvrages : l'un sur l'IA et l'autre sur l'informatique répartie.

Guerraoui R. et Nguang Hoang L., *Turing à la plage. L'intelligence artificielle sur un transat*, Malakoff, Dunod, 2020.

Le second devrait être publié chez Odile Jacob avec Anne-Marie Kermarrec sur le thème du « numérique pour tous ».