

Département d'Informatique  
Faculté des Sciences  
Université de Mons

# Propositions de Mémoires Master 2

## Année 2018-2019

### Contents

1	Modélisation de la déviation d'horloge avec le simulateur Renode	3
2	Synchronisation d'horloges via échange de trames UWB	4
3	Ordonnancement de l'accès au canal de communication dans les réseaux de capteurs sans fil	5
4	Sécurité d'un vote électronique.	6
5	Sécurité du verrouillage central automobile.	7
6	Environnement de gestion automatique des certificats (ACME).	8
7	Confidentialité sur un cloud. <b>Chahrazed BOUDAUD</b>	9
8	<b>Mémoire:</b> Un framework de mutation testing en Python	10
9	<b>Mémoire:</b> Un outil automatique de transformation de statecharts	11
10	<b>Mémoire:</b> Un framework de mutation testing pour statecharts en Sismic	12
11	<b>Mémoire:</b> Process Mining and Comparison of Open Source Software Development Processes	13
12	<b>Mémoire:</b> Apprentissage automatique pour la fusion d'identités des développeurs	15
13	<b>Mémoire:</b> Empirical Analysis of Library Usage in Open Source Software Ecosystems	16
14	<b>Mémoire:</b> Expert recommendation in Question & Answering sites	17
15	Exploitation de données pour accélérer le problème d'isomorphisme de graphes	18
16	AI powered skipper assistant for solar electric yacht	19
17	Database Repairing with Respect to Functional Dependencies	20
18	NULLs and Certain Answers	21

<b>19 Recherche de cycles d'une forme particulière dans les graphes</b>	<b>22</b>
<b>20 Apprentissage passif d'automates</b>	<b>23</b>
<b>21 Apprentissage actif d'automates</b>	<b>24</b>
<b>22 Automates sur mots infinis</b>	<b>25</b>
<b>23 Identification des sommets d'un ensemble de points en présence de bruit et applications</b>	<b>26</b>
<b>24 Analyse d'images hyperspectrales</b>	<b>28</b>
<b>25 Neural Networks and Matrix Factorization</b>	<b>30</b>
<b>26 La factorisation positive de matrices et ses applications</b>	<b>32</b>
<b>27 Regroupement de données dans des sous-espaces linéaires</b>	<b>34</b>
<b>28 La factorisation symétrique et positive de matrices et ses applications</b>	<b>36</b>
<b>29 Partial solvers for parity games: the window approach</b>	<b>38</b>
<b>30 Machine learning for strategy synthesis in Markov decision processes</b>	<b>39</b>
<b>31 Correctness in AI through formal methods</b>	<b>40</b>
<b>32 Development of noise-adaptive learning algorithms.</b>	<b>41</b>

# 1 Modélisation de la déviation d’horloge avec le simulateur Renode

**Service:** Réseaux et Télécommunications

**Directeur:** Bruno Quoitin

**Rapporteurs:** David Hauweele et *autre personne à définir*

## Description

Ce mémoire a pour contexte les *Réseaux de Capteurs sans Fil* ou *Wireless Sensor Networks*. Il s’agit de réseaux interconnectant un grand nombre de noeuds mesurant une ou plusieurs grandeurs physiques dans l’environnement (p.ex. température ou taux d’humidité relatif). Chaque noeud est constitué d’un système embarqué disposant de ressources de traitement, de stockage, de communication et d’énergie limitées. Ces réseaux sont bien étudiés depuis près de deux décennies et une abondante littérature leur est consacrée [1].

L’objectif de ce mémoire est d’étudier le comportement et les performances de ces réseaux par simulation. A cet effet, le simulateur Renode [2, 3] développé par AntMicro sera étudié. Ce simulateur supporte les architectures récentes de noeuds tels que celles reposant sur des processeurs ARM. Le mémoire s’intéressera à la modélisation dans ce simulateur du phénomène de déviation d’horloge entre les noeuds. Ce phénomène a un impact sur la possibilité pour deux ou plusieurs noeuds de communiquer entre eux tout en restant éteints la plupart du temps de façon à conserver leurs ressources énergétiques.

Le mémoire effectuera un état de l’art du problème de la déviation d’horloge [4] et de sa modélisation par simulation [5, 6]. Il proposera ensuite plusieurs approches pour modéliser la déviation d’horloge, en implémentera au moins une qui sera ensuite validée et évaluée expérimentalement.

## Exigences ou prérequis

Intérêt pour les réseaux informatiques et les systèmes embarqués.

## References

- [1] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. “Wireless sensor network survey”. In: *Computer Networks* 52 (12 Aug. 2008), pp. 2292–2330.
- [2] *Renode*. <https://renode.io>.
- [3] *Tutorial: Running Contiki-NG in Renode*. <https://github.com/contiki-ng/contiki-ng/wiki/Tutorial:-Running-Contiki%E2%80%90in-Renode>.
- [4] David W Allan et al. “Time and frequency(time-domain) characterization, estimation, and prediction of precision clocks and oscillators”. In: *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 34.6 (1987), pp. 647–654.
- [5] David Hauweele and Bruno Quoitin. *Toward accurate clock drift modeling in WSN*. Tech. rep. 2018.
- [6] Antoine Fraboulet, Guillaume Chelius, and Eric Fleury. “Worldsens: development and prototyping tools for application specific wireless sensors networks”. In: *Proceedings of the 6th international conference on Information processing in sensor networks*. ACM. 2007, pp. 176–185.

## 2 Synchronisation d’horloges via échange de trames UWB

**Service:** Réseaux et Télécommunications

**Directeur:** Bruno Quoitin

**Rapporteurs:** Maximilien Charlier et *autre personne à définir*

### Description

Il est possible de localiser géographiquement des objets en mesurant le temps de propagation de messages envoyés entre eux ou vers des balises fixes. Le temps de propagation peut ensuite être converti en distance en connaissant la vitesse de propagation. Dans le contexte de ce mémoire, les messages sont envoyés par radio à l’aide de la technologie radio UWB [1] et la vitesse de propagation est celle de la lumière. Etant donnée la grande vitesse de propagation, les temps mesurés sont très courts. A titre d’exemple, une distance d’1 km sera parcourue en  $\sim 3,33\mu s$ . Il est nécessaire pour obtenir une bonne précision de mesure que les horloges des noeuds effectuant la mesure soient synchronisées le plus précisément possible.

L’objectif du projet est de comparer qualitativement et quantitativement plusieurs techniques de synchronisation d’horloge de systèmes informatiques distants. Plusieurs techniques existent telles que le Network Time Protocol (NTP) [2] et la synchronisation via les impulsions reçues toutes les secondes d’un récepteur GPS. Dans un premier temps, le projet documentera le fonctionnement de ces protocoles et techniques. Il en effectuera ensuite une comparaison et discutera des avantages et inconvénients de chaque approche.

Dans un second temps, une mise en oeuvre d’une technique de synchronisation d’horloges issue de la littérature sera effectuée. Il pourrait s’agir par exemple d’une technique basée sur l’échanges de trames UWB et l’usage d’un estimateur appelé *Best Linear Unbiased Estimation* [3]. Les performances de la technique seront évaluées expérimentalement. D’autres estimateurs seront éventuellement utilisés.

### Exigences ou prérequis

Intérêt pour les réseaux informatiques et les systèmes embarqués.

### References

- [1] “IEEE Standard for Low-Rate Wireless Networks”. In: *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)* (Apr. 2016), pp. 1–709. DOI: 10.1109/IEEESTD.2016.7460875.
- [2] Jack Burbank, William Kasch, Professor David L. Mills, and Jim Martin. *Network Time Protocol Version 4: Protocol and Algorithms Specification*. RFC 5905. June 2010. DOI: 10.17487/rfc5905.
- [3] Muhammad Hafeez Chaudhary and Bart Scheers. “Practical One-Way Time Synchronization Schemes With Experimental Evaluation”. In: *Proceedings of the 19th IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. 2018.

### 3 Ordonnancement de l'accès au canal de communication dans les réseaux de capteurs sans fil

**Service:** Réseaux et Télécommunications

**Directeur:** Bruno Quoitin

**Rapporteurs:** Jérémy Dubrulle et Maximilien Charlier

#### Description

Afin d'assurer des communications radio performantes et efficaces énergétiquement dans les réseaux de capteurs sans fil, des recherches récentes ont montré qu'il est bénéfique d'allier multiplexage temporel (TDMA) et changements de canaux radio suivant une séquence pseudo-aléatoire (*channel hopping*). Le multiplexage temporel permet de limiter le nombre de noeuds accédant simultanément au média et par conséquent de meilleures performances d'accès ainsi qu'une consommation énergétique réduite. Les changements de fréquence permettent une meilleure résistance aux interférences. Le protocole TSCH [1] (*Time Slotted Channel Hopping*) standardisé récemment par l'IEEE met en oeuvre ces techniques dans la norme IEEE 802.15.4

La mise en oeuvre de TSCH dans un réseau nécessite l'allocation de paires (slot, canal) aux noeuds. Cette allocation indique à quel moment dans le temps (slot) un noeud (1) doit rester en attente de messages, (2) peut émettre un message ou (3) doit rester endormi. Parallèlement, cette allocation indique pour chaque slot, sur quel canal la communication aura lieu. L'allocation de paires (slot, canal) peut être formulée comme un problème d'optimisation. Il pourrait par exemple s'agir de trouver une allocation qui minimise le temps nécessaire pour l'envoi de messages au travers du réseau.

Ce mémoire s'intéresse de manière générale au problème de l'allocation des slots, aussi appelé ordonnancement des slots. Il s'agira dans un premier temps de comprendre les différentes stratégies d'ordonnancement proposées dans la littérature [2, 3, 4]. Dans un second temps, des stratégies d'ordonnancement pourraient être proposées pour des cas particuliers tels que (1) la géolocalisation de noeuds et (2) des communications associées à différentes qualités de service.

#### Exigences ou prérequis

Intérêt pour les réseaux informatiques et les systèmes embarqués.

#### References

- [1] "IEEE Standard for Low-Rate Wireless Networks". In: *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)* (Apr. 2016), pp. 1–709. DOI: 10.1109/IEEESTD.2016.7460875.
- [2] Rodrigo Hermeto, Antoine Gallais, and Fabrice Theoleyre. "Scheduling for IEEE802.15.4-TSCH and slow channel hopping MAC in low power industrial wireless networks: A survey". In: *Computer Communications* 114 (Oct. 2017).
- [3] Simon Duquennoy, Beshr Al Nahas, Olaf Landsiedel, and Thomas Watteyne. "Orchestra: Robust Mesh Networks Through Autonomously Scheduled TSCH". In: *Proceedings of the International Conference on Embedded Networked Sensor Systems (ACM SenSys 2015)*. Seoul, South Korea, Nov. 2015.
- [4] Tengfei Chang, Mališa Vučinić, Xavier Vilajosana, Simon Duquennoy, and Diego Dujovne. *6TiSCH Minimal Scheduling Function (MSF)*. Internet-Draft draft-chang-6tisch-msf-02. Work in Progress. Internet Engineering Task Force, July 2018. 19 pp. URL: <https://datatracker.ietf.org/doc/html/draft-chang-6tisch-msf-02>.

## 4 Sécurité d'un vote électronique.

**Service:** Réseaux et télécommunications

**Directeur:** A. Buys

**Rapporteurs:** à définir

### Description

Le vote électronique peut-il être sécurisé ? Le mémoire vise à faire le point sur le sujet dans le cadre d'élections traditionnelles (le vote électronique est en passe d'être abandonné en Wallonie pour les élections européennes, fédérales et régionales en Wallonie alors que le gouvernement germanophone opte pour un vote électronique avec confirmation sur papier), mais aussi à caractère plus local (élections de membres de CA, d'assemblées générales d'ASBL, ...) et la possibilité d'organiser un vote via internet. Peut-on assurer l'intégrité du vote, son secret, sa confidentialité ? Des solutions sont actuellement mises en avant, basées sur "bitcoins" et "blockchain" qu'il faudra évaluer dans le cadre de ce mémoire[1].

### Exigences ou prérequis

Un intérêt pour la cryptographie et les réseaux informatiques.

*Ce sujet ne sera pas re-proposé en 2019-2020. A défaut de l'avoir présenté en 2018-2019, l'étudiant devra avoir entamé significativement le travail pour être autorisé à le poursuivre l'année académique suivante.*

### References

- [1] Zhichao Zhao and T-H. Hubert Chan. *How to Vote Privately Using Bitcoin*. <https://eprint.iacr.org/2015/1007.pdf> (visité le 14/06/2017).

## 5 Sécurité du verrouillage central automobile.

**Service:** Réseaux et Télécommunications.

**Directeur:** Alain BUYS

**Rapporteurs:** à définir.

### Description

Les automobiles récentes sont généralement équipées d'un système de verrouillage centralisé : l'utilisateur peut verrouiller ou déverrouiller les portières et/ou le coffre (diverses variantes existent) à l'aide d'un signal radio-fréquence émis à l'aide de boutons sur la clé. Très tôt, il s'est avéré nécessaire d'empêcher les possibles tentatives de rejeu (le signal est intercepté par un attaquant et reproduit à un moment ultérieur). Le système de protection semble assez basique et évite qu'un même code soit réutilisé, via un "roulement" entre une série de codes générés au préalable. Il apparaît cependant que ce système puisse être contourné en associant l'interception à des techniques de brouillage du signal.

Un des buts du projet sera de faire le point sur l'étendue de cette faille et de voir comment ce système pourrait être amélioré.

Il existe une variante "mains libres" au verrouillage classique décrit ci-dessus : il suffit que la clé soit dans le voisinage immédiat du véhicule pour permettre le verrouillage et le déverrouillage via un bouton ou un capteur sur la poignée de porte. Il s'agira de déterminer dans quelle mesure celle-ci est également vulnérable aux attaques évoquées plus haut.

### Références :

<https://assets.documentcloud.org/documents/3010178/Volkswagen-amp-HiTag2-Keyless-Entry-System.pdf>

<https://andrewmohawk.com/2016/02/05/bypassing-rolling-code-systems/> (consulté le 10/7/2018)

<https://conference.hitb.org/hitbsecconf2017ams/sessions/chasing-cars-keyless-entry-system-attacks/> (consulté le 10/7/2018)

### Exigences ou prérequis

Un intérêt pour la cryptographie et les technologies de type radio-fréquence.

*Ce sujet ne sera pas re-proposé en 2019-2020. A défaut de l'avoir présenté en 2018-2019, l'étudiant devra avoir entamé significativement le travail pour être autorisé à le poursuivre l'année académique suivante.*

## 6 Environnement de gestion automatique des certificats (ACME).

**Service:** Réseaux et Télécommunications.

**Directeur:** Alain BUYS

**Rapporteurs:** à définir.

### Description

L'obtention d'un certificat non self-signé nécessite traditionnellement de prouver auprès d'une autorité de certification que l'on est bien détenteur des droits sur le site ou domaine où ce certificat va être utilisé. Cette procédure demande à ce stade un traitement manuel et engendre un coût important.

Des initiatives existent, pour permettre le déploiement automatisé d'une infrastructure à clé publique à très faible coût. L'Internet Security Research Group (ISRG) a créé il y a peu un nouveau protocole ("Automatic Certificate Management Environment") pour son service de création de certificats automatisé "Let's Encrypt". Le but premier de "Let's Encrypt" est d'encourager la conversion de sites web existants de HTTP vers HTTPS. Afin de prouver la propriété du domaine, la technique utilisée actuellement demande que le site concerné soit déjà opérationnel (résolution DNS publique, réponse à des défis sous forme de requêtes HTTP, ...).

Cette procédure de vérification, le rapatriement et l'installation des certificats peuvent être automatisés à l'aide de clients tel que Certbot.

Le projet consistera à faire le point sur le sujet de la certification automatisée en général, notamment en examinant les failles possibles de ces nouveaux systèmes et leurs limitations éventuelles.

Il serait intéressant d'étudier dans quelle mesure les systèmes existants pourraient être adaptés pour une gestion "off-line" de serveurs, de serveurs internes d'une entreprise et notamment l'obtention de certificats utilisés pour d'autres protocoles que HTTP (POP, IMAP, ...).

### Références :

<https://datatracker.ietf.org/doc/draft-ietf-acme-acme/>

<https://letsencrypt.org/>

<https://letsencrypt.org/howitworks/technology/>

<https://certbot.eff.org/docs/intro.html>

### Exigences ou prérequis

Un intérêt pour la cryptographie et les technologies web.

*Ce sujet ne sera pas re-proposé en 2019-2020. A défaut de l'avoir présenté en 2018-2019, l'étudiant devra avoir entamé significativement le travail pour être autorisé à le poursuivre l'année académique suivante.*



## 7 Confidentialité sur un cloud. **Chahrazed BOUDAUD**

**Service:** Réseaux et télécommunications

**Directeur:** A. Buys

**Rapporteurs:** à définir

### Description

De plus en plus de services sont proposés dans le cadre d'un "cloud", services qui ne se limitent pas au stockage des données sur des serveurs externes mais aussi aux différents services dont une entreprise pourrait disposer (gestion d'e-mail, applications web, ...). Cette alternative à la gestion des services sur le propre matériel de l'entreprise peut cependant comporter des risques, notamment en ce qui concerne la confidentialité des données traitées[1]. Le mémoire vise à faire le point sur la question, en particulier sur les possibilités de chiffrement de ces données, proposées par l'hébergeur ou à l'initiative du client.

### Exigences ou prérequis

Un intérêt pour la cryptographie et les réseaux informatiques.

### References

- [1] *Le Cloud Computing. Une opportunité pour l'économie en Belgique.* [http://economie.fgov.be/fr/binaries/20130730\\_Cloud\\_computing\\_FR\\_tcm326-228881.pdf](http://economie.fgov.be/fr/binaries/20130730_Cloud_computing_FR_tcm326-228881.pdf) (visité le 14/06/2017).

## 8 Mémoire: Un framework de mutation testing en Python

**Service:** Service de Génie Logiciel - FS

**Directeur:** Dr. Alexandre Decan ([alexandre.decan@umons.ac.be](mailto:alexandre.decan@umons.ac.be))

**Rapporteurs:** Dr. Tom Mens ([tom.mens@umons.ac.be](mailto:tom.mens@umons.ac.be))

### Description

Le mutation testing est une technique permettant d'évaluer la qualité d'une suite de tests unitaires. Dans les grandes lignes, le mutation testing implique d'automatiquement effectuer de petites modifications dans le code d'un programme. Chaque variation du code est appelée un mutant et est obtenue en appliquant des opérations prédéfinies de mutation (telles qu'inverser un test conditionnel, changer un opérateur booléen ou arithmétique, ...).

Les tests doivent identifier et rejeter ("tuer") ces mutants qui modifient le comportement du programme. La qualité de la suite de tests est alors représentée par la proportion de mutants qui sont tués par les tests. Si un mutant est capable de modifier le comportement du programme sans que la suite de tests ne le détecte, il est alors nécessaire pour le développeur d'ajouter un test afin de détecter cette mutation et de la rejeter.

L'objectif de ce projet est d'implémenter en Python un framework supportant le mutation testing de projets écrits en Python. Ce framework doit permettre (1) de détecter (à l'aide du module "coverage" en Python, par exemple) les parties de code concernées par des tests, (2) d'effectuer des mutations (à l'aide du module "ast" de Python, par exemple) dans ces parties de code, (3) d'exécuter la suite de tests (via les runners de "unittest", par exemple) et (4) de générer un rapport à destination du développeur indiquant quels sont les mutants qui ont été correctement tués par les tests et quels sont ceux qui ne l'ont pas été, afin d'évaluer la qualité de la suite de tests et de suggérer l'écriture de nouveaux tests.

Dans le cadre de ce projet, vous serez amené à collaborer avec un chercheur d'Anvers, expert dans le domaine, et auteur d'un tel outil de mutation testing pour un autre langage. Ce projet vous permettra de (re)découvrir le développement moderne d'applications libres avec Python : le code de l'outil sera distribué sous licence MIT ou LGPL (ou compatible) sur Github, supervisé par un processus d'intégration continu de type Travis-CI. La documentation, au format ReST, sera générée par l'outil Sphinx et distribuée sur ReadTheDocs. La distribution de l'outil sera effectuée sur PyPi (le dépôt accessible via pip).

Pointeurs utiles:

- Mutation testing - [https://en.wikipedia.org/wiki/Mutation\\_testing](https://en.wikipedia.org/wiki/Mutation_testing)
- Coverage - <https://coverage.readthedocs.io/en/coverage-4.4.1/>
- ast - <https://docs.python.org/3/library/ast.html>
- unittest - <https://docs.python.org/3/library/unittest.html>

### Exigences ou prérequis

Une excellente connaissance de programmation en Python 3 est demandé pour mener à bien ce travail.

## 9 Mémoire: Un outil automatique de transformation de statecharts

**Service:** Service de Génie Logiciel - FS

**Directeur:** Dr. Alexandre Decan ([alexandre.decan@umons.ac.be](mailto:alexandre.decan@umons.ac.be))

**Rapporteurs:** Dr. Tom Mens ([tom.mens@umons.ac.be](mailto:tom.mens@umons.ac.be))

### Description

Le **refactoring** est une technique automatique permettant de transformer un programme (ou modèle) dans le but d'améliorer sa qualité, tout en préservant son comportement. Il y a beaucoup de support pour le refactoring au niveau des langages de programmation (surtout les langages orientés objet), mais assez peu au niveau des langages de modélisation.

Les **statecharts** sont un formalisme visuel bien connu pour la modélisation du comportement exécutable de systèmes complexes basés sur des événements. Ils ont été inventés dans les années 80 par David Harel, et sont largement adoptés depuis leur intégration au standard UML. Les statecharts proposent des mécanismes plus avancés que les classiques machines à états. Par exemple, les statecharts supportent la composition hiérarchique d'états, les états parallèles, les gardes (conditions) sur les transitions, etc. Il existe de nombreuses sémantiques et de nombreux outils pour exécuter les statecharts. Sismic est l'un d'eux. Il s'agit d'une librairie Python permettant notamment l'exécution complète d'un statechart défini au format YAML (un format textuel aisément lisible), suivant la sémantique UML/SCXML. La page <http://sismic.readthedocs.org/en/master/format.html#statechart-examples> reprend un exemple de statechart déclaré en YAML, ainsi qu'une représentation visuelle (en PlantUML) de ce même statechart.

Bien que Sismic possède quelques primitives pour modifier des statecharts (ajout ou suppression d'états, renommage d'états, rotation de transitions), il ne permet pas des transformations plus intriquées (comme par exemple: transformer un état en état composite, paralléliser un état composite, ...) ni des opérations plus complexes telles qu'extraire une transition vers un état parent, fusionner des états, etc.

L'objectif de ce projet est de concevoir une extension de Sismic supportant un ensemble d'opérations élémentaires sur les statecharts, ainsi qu'un ensemble de transformations/refactoring non-élémentaires qu'il conviendra d'identifier au préalable. Ces transformations doivent ensuite pouvoir être appliquées sur le statechart (en yaml et/ou en mémoire).

L'étudiant devrait aussi étudier l'état de l'art (et les outils) en matière de refactoring (et plus en particulier le refactoring des modèles statecharts).

Pointeurs utiles:

- Sismic (outil de validation et d'exécution de statecharts) <http://sismic.readthedocs.org/>
- Rope (outil de refactoring et de transformation de code Python) <https://github.com/python-rope/rope>
- YAML (textual markup language) <http://yaml.org/spec/1.1/>
- Yakindu Statechart Tools (un outil de modélisation de statecharts avec support partiel pour le refactoring) <https://www.itemis.com/en/yakindu/state-machine/>

Références:

- Tom Mens, T Tourwé. A survey of software refactoring. IEEE Transactions on software engineering 30 (2), 126-139. 2004
- Martin Fowler. Refactoring: Improving the design of existing code. 1999

### Exigences ou prérequis

Une excellente connaissance de programmation en Python 3, ainsi qu'une bonne connaissance des statecharts en UML est exigé pour mener à bien ce travail.

## 10 **Mémoire:** Un framework de mutation testing pour statecharts en Sismic

**Service:** Service de Génie Logiciel - FS

**Directeur:** Dr. Alexandre Decan ([alexandre.decan@umons.ac.be](mailto:alexandre.decan@umons.ac.be))

**Rapporteurs:** Dr. Tom Mens ([tom.mens@umons.ac.be](mailto:tom.mens@umons.ac.be))

### Description

Les statecharts sont un formalisme visuel bien connu pour la modélisation du comportement exécutable de systèmes complexes basés sur des événements. Ils ont été inventés dans les années 80 par David Harel, et sont largement adoptés depuis leur intégration au standard UML. Les statecharts proposent des mécanismes plus avancés que les classiques machines à états. Par exemple, les statecharts supportent la composition hiérarchique d'états, les états parallèles, les gardes (conditions) sur les transitions, etc.

Il existe de nombreuses sémantiques et de nombreux outils pour exécuter les statecharts. Sismic est l'un d'eux. Il s'agit d'une librairie Python permettant notamment l'exécution complète d'un statechart défini au format YAML (un format textuel aisément lisible), suivant la sémantique UML/SCXML. La page <http://sismic.readthedocs.org/en/master/format.html#statechart-examples> reprend un exemple de statechart déclaré en YAML, ainsi qu'une représentation visuelle (conçue manuellement) de ce même statechart. Sismic fournit aussi du support pour tester les statecharts avec une variante des tests unitaires.

Le mutation testing est une technique permettant d'évaluer la qualité d'une suite de tests unitaires. Dans les grandes lignes, le mutation testing implique d'automatiquement effectuer de petites modifications dans le code d'un programme. Chaque variation du code est appelée un mutant et est obtenue en appliquant des opérations prédéfinies de mutation (telles qu'inverser un test conditionnel, changer un opérateur booléen ou arithmétique, ...). Les tests doivent identifier et rejeter ("tuer") ces mutants qui modifient le comportement du programme. La qualité de la suite de tests est alors représentée par la proportion de mutants qui sont tués par les tests. Si un mutant est capable de modifier le comportement du programme sans que la suite de tests ne le détecte, il est alors nécessaire pour le développeur d'ajouter un test afin de détecter cette mutation et de la rejeter.

L'objectif de ce projet est d'ajouter à Sismic un support pour le mutation testing des statecharts. Ce support doit permettre (1) de détecter les parties de statechart concernées par des tests; (2) d'effectuer des mutations dans ces parties; (3) d'exécuter la suite de tests; et (4) de générer un rapport à destination du développeur indiquant quels sont les mutants qui ont été correctement tués par les tests et quels sont ceux qui ne l'ont pas été, afin d'évaluer la qualité de la suite de tests et de suggérer l'écriture de nouveaux tests.

Pointeurs utiles:

- UML 2.5 - <http://www.omg.org/cgi-bin/doc?formal/15-03-01.pdf>
- Sismic - <http://sismic.readthedocs.org/>
- Mutation testing - [https://en.wikipedia.org/wiki/Mutation\\_testing](https://en.wikipedia.org/wiki/Mutation_testing)
- Coverage - <https://coverage.readthedocs.io/en/coverage-4.4.1/>
- unittest - <https://docs.python.org/3/library/unittest.html>

### Exigences ou prérequis

Une excellente connaissance de programmation en Python 3, ainsi qu'une bonne connaissance des statecharts en UML est exigé pour mener à bien ce travail.

**Service:** Service de Génie Logiciel - FS  
**Directeur:** Dr. Tom Mens (tom.mens@umons.ac.be)  
**Rapporteurs:** *rapporteurs à définir*

```

graph TD
    UNCONFIRMED -- "New bug from a user with a can confirm or a product without UNCONFIRMED state" --> NEW
    UNCONFIRMED -- "Bug confirmed or receives enough votes" --> UNCONFIRMED
    UNCONFIRMED -- "Bug is reopened, was never confirmed" --> UNCONFIRMED
    UNCONFIRMED -- "Developer takes possession" --> ASSIGNED
    NEW -- "Ownership is changed" --> NEW
    NEW -- "Developer takes possession" --> ASSIGNED
    ASSIGNED -- "Development is finished with bug" --> ASSIGNED
    ASSIGNED -- "Development is finished with bug" --> RESOLVED
    RESOLVED -- "Developer takes possession" --> RESOLVED
    RESOLVED -- "Issue is resolved" --> REOPEN
    RESOLVED -- "QA not satisfied with solution" --> REOPEN
    RESOLVED -- "QA verifies solution worked" --> VERIFIED
    REOPEN -- "Bug is reopened" --> UNCONFIRMED
    REOPEN -- "Bug is reopened" --> REOPEN
    REOPEN -- "Bug is reopened" --> VERIFIED
    VERIFIED -- "Bug is closed" --> VERIFIED
    VERIFIED -- "Bug is reopened, was never confirmed" --> UNCONFIRMED
    VERIFIED -- "Bug is closed" --> CLOSED
    CLOSED -- "Bug is reopened, was never confirmed" --> UNCONFIRMED
  
```

Possible resolutions:  
 FIXED  
 DUPLICATE  
 WON'T FIX  
 WORKAROUND  
 INVALID  
 REMIND  
 LATER

Process mining tools (such as Prom and its commercial counterpart Disco) have been developed to facilitate the extraction, analysis and improvement of processes. Among many other features, such tools allow to check whether a given process conforms to a predefined process model, to detect deviations from this model, or even to synthesise process models from analysed process logs.

1. Prom process mining tool <http://www.promtools.org/>
2. Disco process mining tool <http://fluxicon.com/disco/>
3. About the bug/defect life cycle:  
<https://www.bugzilla.org/docs/2.18/html/lifecycle.html>

<http://www.guru99.com/defect-life-cycle.html>  
<http://toolsqa.com/software-testing/defect-life-cycle/>

## 12 **Mémoire:** Apprentissage automatique pour la fusion d'identités des développeurs

**Service:** Service de Génie Logiciel - FS

**Directeur:** Dr. Eleni Constantinou (eleni.constantinou@umons.ac.be) et Dr. Tom Mens (tom.mens@umons.ac.be) et

**Rapporteurs:** à définir

### Description

[Ce mémoire sera à réaliser en anglais.]

Les écosystèmes logiciels open source sont composés de plusieurs milliers de projets développés par plusieurs milliers de développeurs (p.ex., Gnome, Apache, Debian, Ubuntu, CRAN, npm, Eclipse). Pour effectuer des études empiriques de l'évolution d'un tel écosystème, il faut extraire des données historiques provenant des dépôts de code source (p.ex. GitHub ou BitBucket) et d'autres outils utilisés quotidiennement par les développeurs (bug and issue tracker, continuous integration system, mailing list, ...)

Pour analyser les aspects socio-techniques, il est souvent nécessaire de traiter les données historiques des personnes impliquées dans des activités de développement. Puisqu'une personne peut utiliser plusieurs logins et comptes différents (soit dans le même dépôt, ou dans des dépôts différents), il est important d'effectuer une fusion d'identités.

Plusieurs algorithmes et heuristiques ont été proposés pour effectuer une fusion d'identités qui est fiable (en termes de précision et rappel). L'objectif est de proposer des meilleurs techniques de fusion d'identités, en éliminant le plus de faux positifs possible. Les techniques peuvent tenir compte d'avantage d'informations concernant un contributeur, comme le contenu de ses contributions, ses données de géolocalisation, ou encore d'autres informations permettant de mesurer la similarité entre deux contributeurs.

Une piste intéressante à explorer sont les approches d'apprentissage automatique (par exemple basé sur les réseaux de neurones ou des techniques de data mining). Les techniques proposées doivent être appliquées et validées en pratique, et leur fiabilité doit être comparée avec les solutions existantes.

Références utiles:

- C Bird, A Gourley, P Devanbu, M Gertz, A Swaminathan (2006) Mining email social networks ACM International Workshop on Mining software repositories, pages 137-143
- M Goeminne, T Mens (2013) A comparison of identity merge algorithms for software repositories. Science of Computer Programming 78 (8), 971-986
- E Kouters, B Vasilescu, A Serebrenik, MGJ van den Brand (2012) Who's who in Gnome: Using LSA to merge software repository identities. IEEE International Conference on Software Maintenance (ICSM), pages 592-595
- W Mo, B Shen, Y Chen, J Zhu (2015) TBIL: A Tagging-Based Approach to Identity Linkage Across Software Communities. Asia-Pacific Software Engineering Conference (APSEC), pages: 56- 63

## 13 **Mémoire:** Empirical Analysis of Library Usage in Open Source Software Ecosystems

**Service:** Service de Génie Logiciel - FS

**Directeur:** Dr. Tom Mens ([tom.mens@umons.ac.be](mailto:tom.mens@umons.ac.be))

**Rapporteurs:** *rapporteurs à définir*

### Description

When developing software, programmers heavily rely on external software libraries to realise part of the functionality of their application. These libraries can be very diverse and are accessed via their API (Application Programmer Interface), defining the set of features (e.g. classes, interfaces, operation signatures, annotations, etc...) through which the functionalities of the library can be used. Gaining a better understanding of how libraries are effectively used in practice by existing software projects can be very useful to library developers, as it allows them to understand which of the APIs features are used very frequently, and which of them are rarely or never used. Based on this, the API developers can make informed decisions about which changes to make in future versions of their API. For example, API features that are very frequently used should only be changed in backward compatible ways, while rarely used features should be promoted or could be removed or deprecated in newer versions.

The goal of this project is therefore to develop a generic application that takes as input a given (set of) APIs, and a given (set of) open source projects, and that statically analyses the production code and test of the projects to determine how frequently each of the APIs functionalities are accessed, and how this is distributed over the considered projects. As a proof-of-concept, the student should validate the tool he has developed by carrying out an empirical analysis of the usage of a small number of different libraries/APIs by a large number of different open source software projects. The results should be reported to the user in different forms:

- comma-separated value (CSV) files that can easily be processed by statistical analysis tools
- a web-based dashboard that visually displays the API usage statistics
- textual reports in PDF or HTML format

The technical details of how to realise the project are left to the discretion of the student, after agreement of the supervisor. Based on an objective and convincing motivation, the student is free to select :

- the programming language of choice in which to implement his tool.
- the programming language of choice in which the libraries and software projects to be analysed are developed. Popular languages like Java are likely to have a much wider range of libraries available, as well as a wider range of open source projects using them.
- the set of software projects to consider for analysis, and the open source repository from which to extract these projects (and their histories). Existing open source repositories of interest could be, for example, SourceForge or GitHub.
- the set of libraries/APIs to be considered. The more popular APIs are probably more interesting to study, as it will be easier to identify and analyse projects using these APIs.

**[Optional (bonus)]** If time permits, it could be very useful to provide tool support to analyse the evolution of the API usage over time. This involves two dimensions:

1. Study how the use of a given API version changes across successive versions of (a given set of) software projects
2. Study how the API usage evolves over different API versions for a given set of software projects
3. Study how the functionalities of an API evolve over time



## 14 **Mémoire:** Expert recommendation in Question & Answering sites

**Service:** Service de Génie Logiciel - FS

**Directeur:** Dr. Eleni Constantinou (eleni.constantinou@umons.ac.be) and

Dr. Tom Mens (tom.mens@umons.ac.be)

**Rapporteurs:** *rapporteurs à définir*

### Description

While developing software, programmers often seek help by experts in Question & Answering sites like Stack Overflow<sup>1</sup>. In such platforms, developers engage in different activities, like asking questions, answering, commenting on either questions or answers, etc.

Stack Overflow provides mechanisms to motivate users to participate in answering questions in the form of a reputation system (based on Up and Down votes on questions or answers). Stack Overflow is growing fast and a large number of questions is reported daily. This rapid growth makes it challenging to limit the number of questions that remain unanswered. An additional challenge is to show to users posts related to their expertise and interests in order to decrease the number of unanswered questions and at the same time elicit answers quickly. This challenge is also known as the question routing problem.

The goal of this master's thesis is to develop a recommendation mechanism that takes as input users' expertise profiles and given queries (i.e., questions), and it will provide recommendations in the form of which users are the most suitable to answer a question. In practice, this means that Stack Overflow should show the question to those users that are most likely to elicit an answer. To come to such a recommendation system, developer profiles must be built based on users' activity on Stack Overflow, and in a second step these profiles should be enhanced with users' development activity in GitHub. To realise this second step, developer identities must be matched between GitHub and Stack Overflow. The recommendation algorithm will enhance users' expertise in areas closely related to the query while generating recommendations.

To evaluate the recommendation algorithm, already available datasets of Stack Overflow and GitHub will be used. The student will initially setup these datasets and implement the developer matching approach, profile extraction and recommendation algorithm. Next, he/she will evaluate the soundness of the algorithm by using different time thresholds to build developer profiles, and using future data (immediately after each threshold) to validate the accuracy of the recommendation mechanism. The results will be reported for each query, using different recommendation evaluation metrics (such as Ranking Accuracy, Recall, Normalized Discounted Cumulative Gain, etc.).

Prerequisites:

- The student should have excellent programming skills, but the actual programming language being used is free of choice (with the agreement of the supervisors).
- Unit testing: The student must write proper and sufficient unit test to ensure the correctness of the developed tools.
- The student must be good in analytical thinking.

Useful references:

- M. Choetkiertikul, D. Avery, H. K. Dam, T. Tran and A. Ghose (2015) Who Will Answer My Question on Stack Overflow? Australasian Software Engineering Conference, pp. 155-164
- H. Dong, J. Wang, H. Lin, B. Xu and Z. Yang (2015) Predicting Best Answerers for New Questions: An Approach Leveraging Distributed Representations of Words in Community Question Answering. International Conference on Frontier of Computer Science and Technology, pp. 13-18
- Tian, Y., Kochhar, P.S., Lim, E.P., Zhu, F., Lo, D. (2013) Predicting best answerers for new questions: An approach leveraging topic modeling and collaborative voting. Workshops at the International Conference on Social Informatics, pp. 55-68

---

<sup>1</sup><https://stackoverflow.com/>

## 15 Exploitation de données pour accélérer le problème d'isomorphisme de graphes

**Service:** Algorithmique

**Directeur:** H. Mélot

**Rapporteurs:** P. Hauweele et autre personne à définir

### Description

Dans le cadre du développement du logiciel d'aide à la découverte en théorie des graphes PHOEG, le service d'Algorithmique a généré et génère une grande quantité de données relatives aux graphes sous la forme de valeurs d'*invariants* pour ces graphes (comme le diamètre, le nombre chromatique, etc.). Par ailleurs, il est souvent nécessaire dans ce cadre des recherches effectuées dans le service si deux graphes sont isomorphes. Actuellement, le test d'isomorphisme se fait à l'aide de *Nauty*, développé par Brendan McKay.

Le but du mémoire est d'analyser les données (base de données avec les tables d'invariants) pour établir un arbre de décision adapté, prenant en entrée deux graphes, et qui permettrait de rapidement rejeter deux graphes non isomorphes. L'idée étant de ne faire appel à un test d'isomorphisme que dans le cas où cet arbre ne permet pas de rejeter l'isomorphisme. Cet arbre possèdera des noeuds de décisions associés à des invariants qui seront sélectionnés pour leur haute valeur informative. Pour ce faire, l'étudiant devra adapter et étudier les outils permettant de quantifier la valeur informative d'un invariant (par ex. à l'aide de la notion d'entropie).

Une partie importante du mémoire sera consacrée à la validation de l'arbre de décision proposé, sur base des millions de graphes stockés via le système PHOEG.

## 16 AI powered skipper assistant for solar electric yacht

**Service:** Algorithmique

**Directeur:** H. Mélot et S. Dupont

**Rapporteurs:** *A définir*

### Description

The Aquanima 45 is a fully solar powered electric yacht with unrestricted sailing area. Her solar roof harvests solar energy which is stored in a battery bank onboard and managed by a built-in off-the-shelf computer communicating with a tablet via Bluetooth. The propulsion system comprises of 2 separate brushless DC motors controlled manually by the skipper via 2 throttles. The state of charge (SOC) of the batteries and the energy production from the solar roof are shown live on the tablet. The energy consumption varies depending on the boat speed and voyage length. The set speed controls the energy consumption via an exponential relation. It is indispensable to constantly monitor the SOC of charge of the batteries in order to avoid falling short of energy at any time. Therefore, the skipper has to plan the voyage and more concretely adjust the boat speed depending on various factors and adapt the speed along the way. Some of the key factors are:

- Hours of sun exposure per day → Season and latitude
- Sun intensity → Time of the day and latitude
- Cloud coverage → Direct observation and weather forecasts
- Currents/Winds → Direct observation, navigation aids, charts
- Next possible stop, if any

It so happens that the speed regulation is made almost instinctively by the seasoned skipper based on the various inputs available and the voyage plan. A similar electric propulsion system has been modeled in Simulink in the following thesis by Julien Mélot: “Hydrogen / Solar-based Boat Propulsion System: Design, Modelling and Implementation on a Scale Model”. This model can be used as a base to simulate the energy requirements of the Aquanima 45 based on various sailing patterns and voyages. The goal of this thesis is to develop an AI powered skipper assistant to deal with energy consumption.

## 17 Database Repairing with Respect to Functional Dependencies

**Service:** Systèmes d'Information

**Directeur:** Jef Wijsen

**Rapporteurs:**

### Description

Real-world databases are often inconsistent with respect to integrity constraints. The term *database repairing* refers to the process that takes in an inconsistent database, and returns a consistent database that is as similar as possible to the original database. Different similarity measures have been proposed in the literature, giving rise to different repair notions.

The focus of this master thesis is on repairing relational databases with respect to functional dependencies, which are among the most common constraints in the relational model. In a recent article [1], the authors first formalize this problem and then study its complexity. The objective of this master thesis is to study the problem of database repairing with respect to functional dependencies, starting with [1], and to build a prototype to experimentally validate theoretical approaches.

### Exigences ou prérequis

Interest in database theory.

### References

- [1] Ester Livshits, Benny Kimelfeld, and Sudeepa Roy. “Computing Optimal Repairs for Functional Dependencies”. In: *PODS*. ACM, 2018, pp. 225–237.

## 18 NULLs and Certain Answers

**Service:** Systèmes d'Information

**Directeur:** Jef Wijsen

**Rapporteurs:**

### Description

NULL is commonly introduced in the relational model as a placeholder for values that are currently unknown or that do not [yet] exist (e.g., the year of death of a living person). NULLs have been standardized in SQL by means of a three-valued logic (with truth values true, false, and unknown), which has often been criticized for being counter-intuitive.

In some recent articles [1, 2, 3], the authors present new evaluation procedures that bridge the difference between theoretical and practical approaches to answering queries over databases with NULLs. The aim of this master thesis is to study these new evaluation procedures from different viewpoints, including their theoretic complexity as well as their real-life applicability and efficiency.

### Exigences ou prérequis

Interest in database theory.

### References

- [1] Leonid Libkin. “SQL’s Three-Valued Logic and Certain Answers”. In: *ACM Trans. Database Syst.* 41.1 (2016), p. 1. DOI: 10 . 1145 / 2877206. URL: <http://doi.acm.org/10.1145/2877206>.
- [2] Paolo Guagliardo and Leonid Libkin. “Correctness of SQL Queries on Databases with Nulls”. In: *SIGMOD Record* 46.3 (2017), pp. 5–16.
- [3] Paolo Guagliardo and Leonid Libkin. “On the Codd Semantics of SQL Nulls”. In: *AMW*. Vol. 1912. CEUR Workshop Proceedings. CEUR-WS.org, 2017.

## 19 Recherche de cycles d'une forme particulière dans les graphes

**Service:** Informatique Théorique

**Directeur:** Véronique Bruyère

**Rapporteurs:** A définir

### Description

Un *système réactif* est un système plongé dans un environnement, qui doit réagir continuellement aux événements produits par cet environnement. Un exemple est le système ABS qui assiste au freinage chaque fois que celui-ci doit être intense. Une modélisation classique d'un système réactif est celle d'un graphe orienté fini dont les sommets sont répartis en les sommets appartenant au système et les sommets appartenant à l'environnement, et dont les arcs décrivent les interactions entre le système et l'environnement.

Un problème beaucoup étudié est celui de la *synthèse de contrôleur* : on souhaite concevoir un contrôleur qui assure au système réactif de satisfaire une spécification donnée quels que soient les événements produits par l'environnement. Au niveau de la modélisation, la spécification est traduite par une propriété sur le graphe (par exemple atteindre ou éviter un sommet particulier du graphe, passer infiniment souvent par un sommet du graphe, etc.), et concevoir un contrôleur revient à construire une stratégie gagnante du système contre l'environnement pour satisfaire cette propriété.

Dans plusieurs cas de propriétés, l'existence d'une stratégie gagnante dépend de l'existence de certains *cycles* dans le graphe : cycle passant par un sommet d'une certaine forme, cycle de plus petit poids moyen, cycle étiqueté par des tuples de poids tous positifs, etc.

Le but de ce projet est d'étudier en détails plusieurs algorithmes permettant de détecter des cycles d'une forme particulière, et d'implémenter ceux-ci. Une attention particulière sera portée à l'efficacité de ces algorithmes et à la construction de tels cycles.

### Références :

- Richard M Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23:309-311, 1978.
- S. R. Kosaraju and G. F. Sullivan. Detecting cycles in dynamic graphs in polynomial time (preliminary version). In *Proc. of STOC: Symposium on Theory of Computing*, pages 398–406. ACM, 1988.
- Damien Busatto-Gaston, Benjamin Monmege, Pierre-Alain Reynier. Optimal Reachability in Divergent Weighted Timed Games. In *Proc. of FoSSaCS, Lecture Notes in Computer Science 10203*, Springer, 162-178, 2017.

### Remarques :

- Ce sujet convient aussi bien pour un projet de Master 1 que pour un mémoire de Master 2. Son niveau de difficulté et son ampleur seront adaptés en fonction.
- Les étudiants intéressés sont invités à me rencontrer pour discuter plus en détails du sujet.

### Exigences ou prérequis

Goût prononcé pour l'algorithmique et les structures de données avancées.

## 20 Apprentissage passif d'automates

**Service:** Informatique Théorique

**Directeur:** Véronique Bruyère

**Rapporteurs:** A définir

### Description

L'*apprentissage d'automates* trouve de nombreuses applications en traitement de la parole, en traduction automatique, en vérification et synthèse de systèmes informatiques, en biologie computationnelle, en data mining, et même en musique (voir le survey sur le sujet écrit par de la Higuera)

Dans le cadre de ce projet, dans un premier temps, l'étudiant étudiera en profondeur certaines techniques *passives* d'apprentissage d'automates où un automate déterministe de taille minimale est appris automatiquement à partir d'un ensemble donné de mots classés comme appartenant ou n'appartenant pas au langage accepté par l'automate (voir par exemple la thèse de Daniel Neider). Dans un second temps, l'étudiant appliquera ces techniques d'apprentissage au *regular model-checking* (voir à nouveau la thèse de Daniel Neider ainsi que le survey de Parosh Abdulla et al.). Le problème du model-checking consiste à vérifier qu'un système informatique satisfait une spécification quand ceux-ci sont donnés sous la forme de modèles. Des spécifications typiques sont : est-ce que le système n'atteint jamais d'état de deadlock ? Est-ce qu'une requête reçoit toujours une réponse ? De nombreux algorithmes de model-checking ont été élaborés (voir le livre de Baier et Katoen). On parle de "regular" model checking quand le système informatique à vérifier est modélisé en termes d'automates finis.

### Références :

- Colin de la Higuera. A bibliographical study of grammatical inference, Pattern Recognition, 38(9), pp.1332–1348, 2005.
- Daniel Neider, Applications of automata learning in verification and synthesis. PhD thesis, RWTH Aachen University 2014, pp. 1-267.
- Parosh Aziz Abdulla, Bengt Jonsson, Marcus Nilsson, Mayank Saksena: A Survey of Regular Model Checking. CONCUR 2004, Lecture Notes in Computer Science 3170, Springer, pp. 35-48
- Christel Baier, Joost-Pieter Katoen, Principles of model-checking, The MIT Press, 2008.

### Remarques :

- Ce sujet convient aussi bien pour un projet de Master 1 que pour un mémoire de Master 2. Son niveau de difficulté et son ampleur seront adaptés en fonction.
- Les étudiants intéressés sont invités à me rencontrer pour discuter plus en détails du sujet.

### Exigences ou prérequis

Goût prononcé pour l'algorithmique et les structures de données avancées.

## 21 Apprentissage actif d'automates

**Service:** Informatique Théorique

**Directeur:** Véronique Bruyère

**Rapporteurs:** A définir

### Description

L'*apprentissage d'automates* trouve de nombreuses applications en traitement de la parole, en traduction automatique, en vérification et synthèse de systèmes informatiques, en biologie computationnelle, en data mining, et même en musique (voir le survey sur le sujet écrit par de la Higuera).

Angluin a introduit les bases de l'apprentissage *actif* d'automates dans son célèbre article de 1987: un automate est appris en interrogeant un professeur qui connaît l'automate  $\mathcal{A}$ : soit en lui demandant si un mot donné est accepté par l'automate  $\mathcal{A}$ , soit en lui demandant si l'automate appris est équivalent à l'automate  $\mathcal{A}$ . Cet article est à la base de nombreux travaux sur l'apprentissage actif (voir par exemple le texte de Frits W. Vaandrager); des bibliothèques ont été développées comme *LearnLib* (<https://learnlib.de/>).

Le but de ce mémoire est de comprendre certains algorithmes d'apprentissage actif d'automates, et ensuite de les appliquer dans des cas concrets en utilisant la bibliothèque LearnLib.

### Références :

- Colin de la Higuera. A bibliographical study of grammatical inference, Pattern Recognition, 38(9), pp.1332–1348, 2005.
- Dana Angluin. Learning regular sets from queries and counterexamples. Inf. Comput. 75, 2 (1987), 87–106.
- Frits W. Vaandrager. Model learning. Commun. ACM 60(2): 86-95 (2017)

### Remarque :

- Les étudiants intéressés sont invités à me rencontrer pour discuter plus en détails du sujet.

### Exigences ou prérequis

Un intérêt pour la théorie des automates et ses extensions, ainsi que pour les structures de données et les algorithmes. La capacité de lire des articles scientifiques d'informatique théorique.



## 22 Automates sur mots infinis

**Service:** Informatique Théorique

**Directeur:** Véronique Bruyère

**Rapporteurs:**

### Description

Les automates acceptant des mots infinis sont une extension des automates acceptant des mots finis. Ces automates ainsi que les langages  $\omega$ -réguliers qu'ils acceptent ont été beaucoup étudiés et sont utilisés dans divers domaines comme par exemple le "model-checking". Récemment, les auteurs de [1] ont proposé un modèle équivalent aux automates sur mots infinis, qui est composé d'une famille finie d'automates acceptant des mots finis. Il s'en suit des algorithmes simples pour effectuer les opérations booléennes sur les langages  $\omega$ -réguliers, pour tester l'équivalence de deux automates sur mots infinis, ou l'appartenance d'un mot infini à un langage  $\omega$ -régulier.

Un autre article récent [2] exploite ce modèle pour faire de l'apprentissage d'automate sur mots infinis. Dans ce contexte, un élève souhaite apprendre la structure d'un automate  $\mathcal{A}$  connu d'un professeur. L'élève pose des questions au professeur du type : tel mot est-il accepté ou pas par  $\mathcal{A}$  ? tel automate est-il équivalent à l'automate  $\mathcal{A}$  ? L'article [2] propose un algorithme d'apprentissage de l'automate  $\mathcal{A}$  basé sur certains algorithmes de l'article [1].

Une première phase du mémoire est de comprendre le nouveau modèle d'automate et les algorithmes proposés dans [1] dans le but d'implémenter ceux-ci. Une seconde phase est de comprendre l'algorithme d'apprentissage de [2], de l'implémenter et de reproduire certaines expérimentations de cet article.

**Remarque :**

- Les étudiants intéressés sont invités à me rencontrer pour discuter plus en détails du sujet.

### Exigences ou prérequis

Un intérêt pour la théorie des automates et ses extensions, ainsi que pour les structures de données et les algorithmes. La capacité de lire des articles scientifiques d'informatique théorique.

### References

- [1] Dana Angluin, Udi Boker, and Dana Fisman. "Families of DFAs as Acceptors of omega-Regular Languages". In: *41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland*. Ed. by Piotr Faliszewski, Anca Muscholl, and Rolf Niedermeier. Vol. 58. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, 11:1–11:14. ISBN: 978-3-95977-016-3. DOI: 10.4230/LIPIcs.MFCS.2016.11. URL: <https://doi.org/10.4230/LIPIcs.MFCS.2016.11>.
- [2] Dana Angluin and Dana Fisman. "Learning regular omega languages". In: *Theor. Comput. Sci.* 650 (2016), pp. 57–72. DOI: 10.1016/j.tcs.2016.07.031. URL: <https://doi.org/10.1016/j.tcs.2016.07.031>.

## **23 Identification des sommets d'un ensemble de points en présence de bruit et applications**

**Service:** Mathématique et Recherche opérationnelle (FPMs)

**Directeur:** Nicolas Gillis

**Rapporteurs:** à *définir*

### **Description**

Voir la page suivante.

**Sujet :** Identification des sommets d'un ensemble de points en présence de bruit et applications

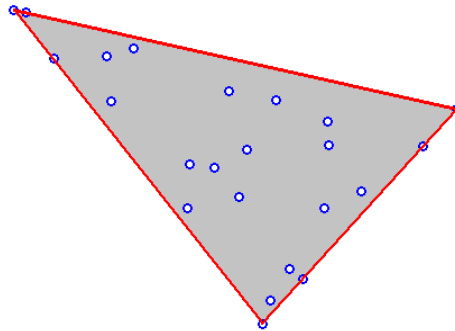
**Section :** IG

**Encadrement :** Nicolas Gillis (FPMs)

Dans ce TFE, on souhaite étudier le problème géométrique suivant : étant donné un ensemble  $X$  de  $n$  points en dimension  $d$  :

$$X = \{ x_i \in \mathbb{R}^d, 1 \leq i \leq n \},$$

on désire trouver un sous-ensemble  $S$  de points de  $X$  tel que tous les autres points de  $X$  peuvent s'écrire comme une combinaison convexe de points de  $S$  (une combinaison convexe est une combinaison linéaire dont les poids sont positifs et dont la somme des poids vaut un). Par exemple, en deux dimensions ( $d=2$ ), cela revient à identifier les sommets du polygone engendré par  $X$ .



**Exemple :** Pour cet ensemble de points qui génère un triangle, trois points suffisent pour représenter, via une combinaison convexe, n'importe quel autre point.

Le problème d'identifier ces sommets est relativement simple à résoudre. Par exemple, on peut rapidement tester, pour chaque point, s'il peut être représenté ou non comme une combinaison convexe des autres (il s'agit de résoudre un système linéaire d'égalités et d'inégalités).

Seulement, en pratique, les points sont souvent perturbés avec du bruit (par exemple dû à des erreurs de mesures, des mauvaises conditions, et des données manquantes) et il est dès lors très important en pratique d'être capable d'identifier l'ensemble  $S$  quand du bruit est présent.

L'objectif de ce TFE sera de mettre au point des algorithmes efficaces pour identifier  $S$ , même en présence de bruit, de les comparer et enfin de les utiliser dans diverses applications, en particulier pour l'analyse :

- de documents : le but est d'identifier les différents sujets traités par les documents.
- d'images hyperspectrales : le but est d'identifier les matériaux présents dans l'image.

## **24 Analyse d'images hyperspectrales**

**Service:** Mathématique et Recherche opérationnelle (FPMs)

**Directeur:** Nicolas Gillis

**Rapporteurs:** à *définir*

### **Description**

Voir la page suivante.

**Sujet :** Analyse d'images hyperspectrales

**Section :** IG, Elec

**Encadrement :** Nicolas Gillis (FPMs)

Une image couleur contient l'information correspondant à trois longueurs d'ondes du spectre visible (rouge – 650nm, vert – 550nm, et bleu – 450nm). Les images hyperspectrales comportent davantage de longueurs d'ondes, typiquement une centaine, couvrant en général les longueurs d'onde entre 400 et 2500nm. Ces images permettent donc d'identifier des détails invisibles à l'œil nu et sont utilisées dans de nombreuses applications : industrie alimentaire (contrôle de qualité), agriculture (suivi du développement et de la santé des cultures), minéralogie et chimie (composition et concentration de composants et réactions chimiques), environnement (suivi des sources de pollution), imagerie médicale (détection de tumeur), militaire (détection de camouflage); voir par exemple [http://en.wikipedia.org/wiki/Hyperspectral\\_imaging](http://en.wikipedia.org/wiki/Hyperspectral_imaging) pour plus d'informations. Ce domaine est en pleine expansion grâce au développement récent de ce type de caméras.

Etant donné une telle image, une tâche cruciale est d'extraire l'information importante : identifier les matériaux que cette image contient et classifier les pixels (c'est-à-dire évaluer quels matériaux chaque pixel contient et en quelles quantités) ; voir ci-dessous pour une illustration.

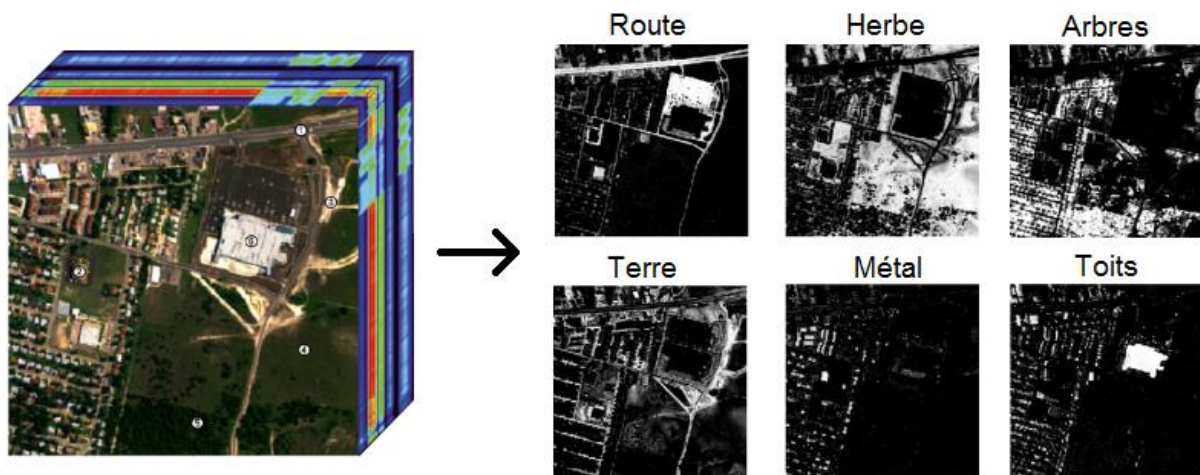


Figure 1. Décomposition d'une image hyperspectrale

Dans ce travail de fin d'étude, plusieurs aspects peuvent être envisagés :

- Étude théorique de modèles physiques et mathématiques pour l'analyse d'images hyperspectrales. Ces modèles sont liés à des problèmes fondamentaux tels que la factorisation matricielle ou l'identification des sommets d'un polytope. Ces modèles permettent la séparation de sources et ont d'autres applications (documents, sons, etc.).
- Développement et comparaison d'algorithmes pour la classification.
- Développement d'une toolbox pour l'analyse d'images hyperspectrales.

## 25 Neural Networks and Matrix Factorization

**Service:** Mathématique et Recherche opérationnelle (FPMs)

**Directeur:** Nicolas Gillis

**Rapporteurs:** à *définir*

### Description

Voir la page suivante.

**Sujet:** Neural Networks and Matrix Factorization

**Section:** IG

**Ecadrement:** Nicolas Gillis, Arnaud Vandaele, Xavier Siebert (FPMs)

Let  $x_i \in \mathbb{R}^{d_x}$  be  $n$  input data points ( $1 \leq i \leq n$ ) in dimension  $d_x$ , and  $y_i \in \mathbb{R}^{d_y}$  be the corresponding output labels in dimension  $d_y$ . We will denote  $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{d_x \times n}$  the matrix containing the  $n$  data points and similarly for  $Y = [y_1, y_2, \dots, y_n] \in \mathbb{R}^{d_y \times n}$ . In this work, we will first consider a neural network with one hidden layer of dimension  $d_h$  in order to classify the data points  $x_i$ 's according to their labels  $y_i$ 's; see Figure 1 for an illustration. Note that if  $X = Y$  then we have a so-called *auto-encoder* neural network which is useful to extract features automatically within the hidden layers.

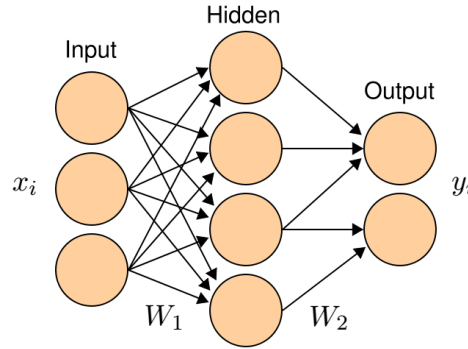


Figure 1: Illustration of a one-hidden-layer neural network, with  $d_x = 3$ ,  $d_h = 4$  and  $d_y = 2$ .

Defining the weights between the input and hidden layers as  $W_1 \in \mathbb{R}^{d_h \times d_x}$ , the hidden variables are given by  $W_1 X$ . In the hidden neurons, a non-linear function is applied to the hidden variables that we denote  $h(\cdot)$ . We also define  $W_2 \in \mathbb{R}^{d_y \times d_h}$  as the weights between the hidden and output layers so that, in an ideal situation (no noise, correct labels), we have

$$Y = W_2 h(W_1 X).$$

Note that for simplicity we have not considered in this model possible bias, but they can easily be taken into account (using  $Y = W_2 h(W_1 X + B_1) + B_2$  where the columns  $B_1$  are equal to one another, and similarly for  $B_2$ ).

We will consider rectified linear unit (ReLU) as the non-linear function  $h$ , that is,  $h(x) = \max(0, x) = [x]_+$  (negative entries are set to zero). In practice, one has to compute the matrices  $W_1$  and  $W_2$  in order to minimize some loss function. Using least squares, we obtain the following optimization problem

$$\min_{W_1, W_2} \|Y - W_2 [W_1 X]_+\|_F^2 \quad (1)$$

## **26 La factorisation positive de matrices et ses applications**

**Service:** Mathématique et Recherche opérationnelle (FPMs)

**Directeur:** Nicolas Gillis

**Rapporteurs:** à *définir*

### **Description**

Voir la page suivante.



**Sujet :** La factorisation positive de matrices et ses applications

**Section :** IG

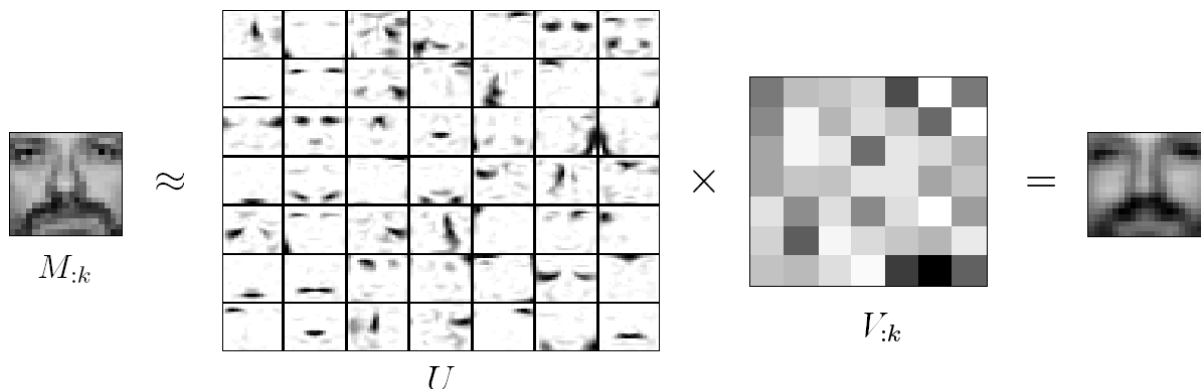
**Encadrement :** Nicolas Gillis, Arnaud Vandaele (FPMs)

Ce travail de fin d'étude a pour objet le problème de la factorisation positive de matrices (en anglais : nonnegative matrix factorization -- NMF) qui peut être défini comme suit : étant donné une matrice  $M$  positive (c'est-à-dire une matrice dont toutes les entrées sont positives et on écrira  $M \geq 0$ ) ayant  $p$  lignes et  $n$  colonnes et un rang de factorisation  $r \ll \min(p,n)$ , on désire trouver deux matrices  $U$  ( $p$  lignes,  $r$  colonnes) et  $V$  ( $r$  lignes,  $n$  colonnes) également positives et telles que

$$M \approx UV.$$

Une première application de la NMF est la compression de  $M$  : en effet, si  $r$  est suffisamment petit (précisément, si  $r < \frac{np}{n+p}$ ), le nombre d'entrées dans  $U$  et  $V$  ( $=pr+rn$ ) est plus petit que le nombre d'entrées dans  $M$  ( $=np$ ).

En analyse d'images, la NMF permet d'extraire automatiquement des caractéristiques communes et localisées de ces images ; voir ci-dessous où la NMF a été appliquée à un ensemble d'images de visages.



The diagram shows the equation  $M_{:k} \approx U V_{:k}$ . On the left is a grayscale face image labeled  $M_{:k}$ . This is followed by an approximation symbol  $\approx$ . Then is a matrix  $U$ , represented as an 8x8 grid of small grayscale face patches. This is followed by a multiplication symbol  $\times$ . Then is a vector  $V_{:k}$ , represented as an 8x1 column of grayscale squares. This is followed by an equals symbol  $=$ . On the right is the reconstructed grayscale face image.

En analyse de textes, la NMF permet d'identifier automatiquement différentes catégories/sujets (c'est-à-dire des ensembles de mots qui apparaissent simultanément dans un sous ensemble de textes) et de classifier les textes dans ces différentes catégories.

Dans ce mémoire, il y aura la possibilité d'étudier différents aspects de ce problème :

- (*Algorithmes*) Mise au point d'algorithmes efficaces pour calculer des factorisations positives ( $U, V \geq 0$ ) étant donné une matrice  $M \geq 0$ .
- (*Applications*) Utiliser la NMF pour différentes applications, et la comparer à d'autres techniques existantes.
- (*Théorie*) Malgré que la NMF soit un problème très difficile, il est possible de mettre au point des algorithmes efficaces dans certains cas particuliers, et de prouver leur efficacité.

## **27 Regroupement de données dans des sous-espaces linéaires**

**Service:** Mathématique et Recherche opérationnelle (FPMs)

**Directeur:** Nicolas Gillis

**Rapporteurs:** à *définir*

### **Description**

Voir la page suivante.

**Sujet :** Regroupement de données dans des sous-espaces linéaires

**Section :** IG

**Encadrement :** Nicolas Gillis (FPMs)

Le problème de regroupement (*clustering*) a pour objectif de classer des données dans différents ensembles (on supposera que chaque élément de ces données est représenté par un vecteur de  $\mathbb{R}^m$ ). Par exemple, le clustering permet de classer un ensemble de documents en fonction du sujet traité, ou encore de classer des images en fonction de leur type (paysage, portrait, etc.) ou de ce qu'elles contiennent (p.ex., chien vs. chat, motos vs. voitures, etc.).

Dans ce travail, on propose de se concentrer sur un problème de clustering particulier : on va supposer que les éléments d'un même sous-ensemble appartiennent à un sous-espace linéaire de faible dimension (ce problème est appelé *subspace clustering* en anglais). En termes mathématiques, si  $x_1, x_2, \dots, x_n \in \mathbb{R}^m$  sont des points appartenant au même sous-ensemble, alors on va supposer qu'il existe un petit ensemble de vecteurs  $y_1, y_2, \dots, y_r \in \mathbb{R}^m$  (où  $r \ll n$ ) tels que pour tout  $1 \leq i \leq n$  :

$$x_i = \sum_{k=1}^r \alpha_i(k) y_k + n_i \quad \text{pour des poids appropriés } \alpha_i \in \mathbb{R}^r, \text{ et du bruit } n_i \in \mathbb{R}^m.$$

Dans ce mémoire, on propose d'étudier différents algorithmes pour résoudre ce problème. En fonction des préférences de l'étudiant, ces algorithmes pourront être basés sur l'optimisation convexe (en particulier, l'optimisation linéaire) et/ou sur des métaheuristiques (optimisation combinatoire). On appliquera ces algorithmes à des problèmes de classifications de documents et d'images; voir ci-dessous pour deux exemples d'applications<sup>1</sup>.



Fig. 1. Motion segmentation: given feature points on multiple rigidly moving objects tracked in multiple frames of a video (top), the goal is to separate the feature trajectories according to the moving objects (bottom).



Fig. 2. Face clustering: given face images of multiple subjects (top), the goal is to find images that belong to the same subject (bottom).

---

<sup>1</sup> Images provenant de l'article : Elhamifar and Vidal, *Sparse Subspace Clustering: Algorithm, Theory, and Applications*, IEEE Trans. on Pattern Analysis and Machine Intelligence 35(11): 2765-2781, 2013.

## **28 La factorisation symétrique et positive de matrices et ses applications**

**Service:** Mathématique et Recherche opérationnelle (FPMs)

**Directeur:** Nicolas Gillis

**Rapporteurs:** à *définir*

### **Description**

Voir la page suivante.

**Sujet :** La factorisation symétrique et positive de matrices et ses applications

**Section :** IG

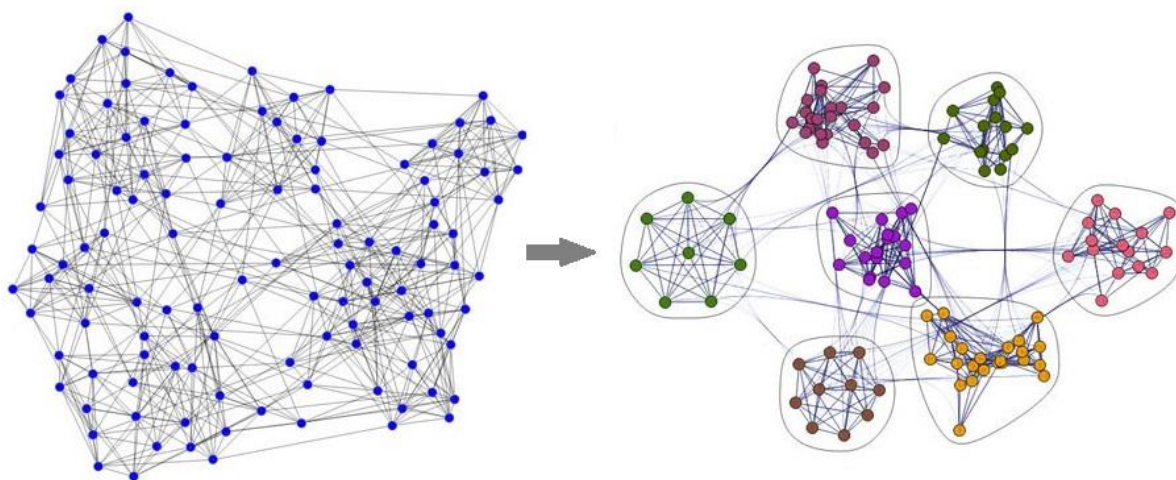
**Encadrement :** Nicolas Gillis, Arnaud Vandaele (FPMs)

Ce travail de fin d'étude a pour objet le problème de la factorisation symétrique positive de matrices (en anglais : symmetric nonnegative matrix factorization -- symNMF) qui peut être défini comme suit : étant donné une matrice  $M$  positive et symétrique (c'est-à-dire une matrice dont toutes les entrées sont positives et on écrira  $M \geq 0$ , et  $M_{ij} = M_{ji}$  pour tout  $i, j$ ) ayant  $n$  lignes et  $n$  colonnes et un rang de factorisation  $r$ , on désire trouver une matrice  $H$  ( $n$  lignes,  $r$  colonnes) également positive et telle que

$$M \approx H H^T.$$

La symNMF peut être utilisée pour faire du clustering : En définissant chaque entrée  $M_{ij}$  de  $M$  comme une mesure de similarité entre l'objet  $i$  et l'objet  $j$ , on peut montrer que chaque colonne de la matrice  $H$  va correspondre, automatiquement, à des groupes d'objets partageant des similarités.

Par exemple, dans l'analyse de graphe (un graphe peut par exemple représenter un réseau social tel que Facebook, avec  $M_{ij} = 1$  si et seulement si les personnes  $i$  et  $j$  sont amies,  $M_{ij} = 0$  sinon), cette technique permet d'automatiquement extraire des communautés de personnes (c.-à.-d. des sous-ensembles de personnes très connectées). Sur l'image ci-dessous, on a le graphe original à gauche, et 7 communautés identifiées à droite.



En analyse de textes, cette technique permet d'automatiquement regrouper des textes discutant de sujets similaires.

Dans ce mémoire, il y aura la possibilité d'étudier différents aspects de ce problème :

- (*Algorithmes*) Mise au point d'algorithmes efficaces basés sur des heuristiques pour calculer des solutions  $H \geq 0$  de la symNMF.
- (*Applications*) Utiliser la symNMF pour différentes applications, et la comparer à d'autres techniques existantes.

## 29 Partial solvers for parity games: the window approach

**Service:** Mathématiques Effectives.

**Directeur:** Mickael Randour (co-directeur possible).

**Rapporteurs:** à définir.

### Description

*Parity games* are a core mathematical model underlying many techniques in the formal verification and automated synthesis of provably-correct computer systems. They are two-player zero-sum turn-based games played on graphs [1]. Given a starting vertex, one of the player necessarily has a winning strategy, as such games are determined. Deciding who has a winning strategy is a canonical problem for the complexity class  $NP \cap coNP$  and despite continuous effort, no polynomial algorithm has been found yet [2].

To provide efficient software tools despite this barrier, researchers have developed *partial solvers* for parity games: algorithms that provide correct answers but may not be complete (with regard to vertices of the graph). The goal of this project is to study such approaches (e.g., [3, 4]) and assess the applicability of *window parity games* [5] as a new mechanism for partial solvers.

### Exigences ou prérequis

Basic notions of algorithmics and programming.

### References

- [1] Erich Grädel, Wolfgang Thomas, and Thomas Wilke, eds. *Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001]*. Vol. 2500. Lecture Notes in Computer Science. Springer, 2002. ISBN: 3-540-00388-6. DOI: 10.1007/3-540-36387-4. URL: <https://doi.org/10.1007/3-540-36387-4>.
- [2] Cristian S. Calude, Sanjay Jain, Bakhadyr Khoussainov, Wei Li, and Frank Stephan. “Deciding parity games in quasipolynomial time”. In: *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*. Ed. by Hamed Hatami, Pierre McKenzie, and Valerie King. ACM, 2017, pp. 252–263. ISBN: 978-1-4503-4528-6. DOI: 10.1145/3055399.3055409. URL: <http://doi.acm.org/10.1145/3055399.3055409>.
- [3] Steen Vester. “Winning Cores in Parity Games”. In: *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS ’16, New York, NY, USA, July 5-8, 2016*. Ed. by Martin Grohe, Eric Koskinen, and Natarajan Shankar. ACM, 2016, pp. 662–671. ISBN: 978-1-4503-4391-6. DOI: 10.1145/2933575.2933589. URL: <http://doi.acm.org/10.1145/2933575.2933589>.
- [4] Michael Huth, Jim Huan-Pu Kuo, and Nir Piterman. “Static Analysis of Parity Games: Alternating Reachability Under Parity”. In: *Semantics, Logics, and Calculi - Essays Dedicated to Hanne Riis Nielson and Flemming Nielson on the Occasion of Their 60th Birthdays*. Ed. by Christian W. Probst, Chris Hankin, and René Rydhof Hansen. Vol. 9560. Lecture Notes in Computer Science. Springer, 2016, pp. 159–177. ISBN: 978-3-319-27809-4. DOI: 10.1007/978-3-319-27810-0\_8. URL: [https://doi.org/10.1007/978-3-319-27810-0\\_8](https://doi.org/10.1007/978-3-319-27810-0_8).
- [5] Véronique Bruyère, Quentin Hautem, and Mickael Randour. “Window parity games: an alternative approach toward parity games with time bounds”. In: *Proceedings of the Seventh International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2016, Catania, Italy, 14-16 September 2016*. Ed. by Domenico Cantone and Giorgio Delzanno. Vol. 226. EPTCS. 2016, pp. 135–148. DOI: 10.4204/EPTCS.226.10. URL: <https://doi.org/10.4204/EPTCS.226.10>.

## 30 Machine learning for strategy synthesis in Markov decision processes

**Service:** Mathématiques Effectives.

**Directeur:** Mickael Randour (co-directeur possible).

**Rapporteurs:** à définir.

### Description

*Markov decision processes* (MDPs) are widely used to reason about decision-making in stochastic environments (e.g., journey planning [1]). Efficient algorithms exist to compute optimal control strategies, and have been implemented in successful tools such as Storm (<http://www.stormchecker.org/>) or PRISM (<https://www.prismmodelchecker.org/>). Still, they fail when considering very large MDPs that arise naturally in practice.

Recently, approaches have been developed to combine machine learning and formal methods in a way that permits to synthesize strategies with provable guarantees with largely increased efficiency (e.g., [2]). The goal of this project is to study such methods and to try to extend them to *multi-objective* models.

### Exigences ou prérequis

Basic notions of probabilities, algorithmics, and programming.

### References

- [1] Mickael Randour, Jean-François Raskin, and Ocan Sankur. “Variations on the Stochastic Shortest Path Problem”. In: *Verification, Model Checking, and Abstract Interpretation - 16th International Conference, VMCAI 2015, Mumbai, India, January 12-14, 2015. Proceedings*. Ed. by Deepak D’Souza, Akash Lal, and Kim Guldstrand Larsen. Vol. 8931. Lecture Notes in Computer Science. Springer, 2015, pp. 1–18. ISBN: 978-3-662-46080-1. DOI: 10.1007/978-3-662-46081-8\_1. URL: [https://doi.org/10.1007/978-3-662-46081-8%5C\\_1](https://doi.org/10.1007/978-3-662-46081-8%5C_1).
- [2] Tomas Brazdil, Krishnendu Chatterjee, Martin Chmelik, Vojtech Forejt, Jan Kretinsky, Marta Z. Kwiatkowska, David Parker, and Mateusz Ujma. “Verification of Markov Decision Processes Using Learning Algorithms”. In: *Automated Technology for Verification and Analysis - 12th International Symposium, ATVA 2014, Sydney, NSW, Australia, November 3-7, 2014, Proceedings*. Ed. by Franck Cassez and Jean-François Raskin. Vol. 8837. Lecture Notes in Computer Science. Springer, 2014, pp. 98–114. ISBN: 978-3-319-11935-9. DOI: 10.1007/978-3-319-11936-6\_8. URL: [https://doi.org/10.1007/978-3-319-11936-6%5C\\_8](https://doi.org/10.1007/978-3-319-11936-6%5C_8).

## 31 Correctness in AI through formal methods

**Service:** Mathématiques Effectives.

**Directeur:** Mickael Randour (co-directeur possible).

**Rapporteurs:** à définir.

### Description

The ever-increasing resort to learning and AI in our lives has led to concerns regarding their safety. Obviously, researchers in the field did not wait for formal methods to reason about the correctness of learners. Core concepts such as **probably approximately correct (PAC) learning** have been around for decades [1]. Their overall objective is to guarantee that with high probability, the approximation granted by the learner will converge to a solution close to the optimum.

As formal methods get closer to learning, we hope to go further and provide strong guarantees (e.g., worst-case) on learning models and AI frameworks, notably based on concepts of probabilistic model checking. The goal of this project is to establish a comparative overview of correctness measures in AI and formal methods and to draw the first lines of an alliance between both fields.

### Exigences ou prérequis

Basic notions of probabilities, algorithmics, and programming.

### References

- [1] Leslie G. Valiant. “A Theory of the Learnable”. In: *Commun. ACM* 27.11 (1984), pp. 1134–1142. DOI: 10.1145/1968.1972. URL: <http://doi.acm.org/10.1145/1968.1972>.



## 32 Development of noise-adaptive learning algorithms.

**Service:** Mathématique et Recherche Opérationnelle (FPMs) / Mathématiques Effectives (FSc)

**Directeur:** Xavier Siebert (**co-Directeur** : Mickael Randour).

**Rapporteurs:** à définir

### Description

Keywords : computational learning theories, machine learning, classification, risk bounds

The overall aim of computational learning theories is to provide guarantees about the convergence of learning algorithms, usually expressed in terms of the number of data points required to reach a given level of accuracy [1].

In particular, we are interested in active learning algorithms, where the learner can interactively request labels at any point in the data space to speedup learning. Recent work in this field has provided algorithms with strong theoretical properties, including the adaptivity to the level of noise in the data [2]. However, these algorithms are usually not practical to implement, especially in the case of multi-dimensional data.

The objective of this work is thus to provide algorithms that are practical to use, while preserving their theoretical properties.

### Exigences ou prérequis

Basic notions of probabilities, algorithmics, and programming.

### References

- [1] Olivier Bousquet, Stéphane Boucheron, and Gábor Lugosi. “Introduction to statistical learning theory”. In: *Advanced lectures on machine learning*. Springer, 2004, pp. 169–207.
- [2] Andrea Locatelli, Alexandra Carpentier, and Samory Kpotufe. “Adaptivity to noise parameters in nonparametric active learning”. In: *arXiv preprint arXiv:1703.05841* (2017).