

# Audition pour le poste de maître de conférences numéro 251816, section 27

ENSEIRB-MATMECA (Bordeaux INP) – LaBRI

Gaëtan Staquet

Centre Inria de l'Université de Rennes

6 mai 2025

Pour consulter ces diapositives :

<https://www.gaetanstaquet.com/MCF/AuditionBordeauxINP.pdf>

# Qui suis-je ?

## Parcours académique :

**2024 – Présent** Postdoctorat au Centre Inria de l'Université de Rennes.

Collaboration : Nathalie Bertrand et Patricia Bouyer (LMF, ENS Paris-Saclay).

**2020 – 2024** Doctorat à l'Université de Mons et l'Université d'Anvers.

Supervision : Véronique Bruyère et Guillermo A. Pérez.

**Août 2023 – Oct. 2023** Séjour de recherche à l'Université Radboud (Nijmegen, Pays-Bas) chez Frits W. Vaandrager.

**2015 – 2020** Bachelier et master en sciences informatiques à l'Université de Mons.

**Thématique de recherche** : Apprentissage et vérification de modèles pour des systèmes.

## Participation aux activités collectives :

- ▶ Membre de plusieurs Conseils et Comités.
- ▶ Revues pour 2 journaux et 5 conférences.
- ▶ Activités de vulgarisation.

**Enseignements** : 251h équivalent TD.

## Nouveaux éléments :

- ▶ 2 articles soumis pour relecture aux conférences MFCS et QEST+FORMATS.
- ▶ Comité d'évaluation des artefacts pour QEST+FORMATS 2025.

# Mon expérience d'enseignement – 251h équivalent TD

## Programmation et Algorithmique :

- ▶ **Typologie des langages**, L3 (EPITA), 8h CM et 4h TD, 2024 – 2025.
- ▶ **Algorithmique et Complexité Expérimentale**, L1 (ISTIC), 28,5h TP, 2024 – 2025.
- ▶ **Programmation Logique** (Prolog) et **Programmation Fonctionnelle** (Scheme), L3 (UMONS), 42h et 48h TD, 2021 – 2022 à 2023 – 2024.
- ▶ **Programmation et Algorithmique I** (Python) et **Programmation et Algorithmique II** (Java), L1 (UMONS), 80h et 80h TP, 2018 – 2019 et 2019 – 2020.

## Intelligence artificielle :

- ▶ **Intelligence artificielle et Machine Learning**, L3 (EPITA), CM et TD, 2024 – 2025.

## Implication personnelle :

- ▶ Préparer et corriger des examens.
- ▶ Co-encadrer des mémoires de master.
- ▶ Encadrement de 5 stagiaires.
- ▶ Évaluer et assister aux soutenances de 11 mémoires de master.

## Enseigner sur les trois années et dans les deux formations :

- ▶ Informatique théorique ;
- ▶ Programmation et algorithmique ;
- ▶ Mathématiques pour l'informatique.
- ▶ Intelligence artificielle ;
- ▶ Génie logiciel ;

## Me former pour élargir mes possibilités d'enseignement :

**Moyen terme** Systèmes d'exploitation ; Compilation ; Programmation système.

**Long terme** Robotique ; FabLab.

## Prendre des responsabilités :

- ▶ Encadrement d'élèves.
- ▶ Suivis de stage.
- ▶ Implication dans la vie de l'ENSEIRB-MATMECA.

## Féminisation et inclusion :

- ▶ Rendre les études plus visibles (**Moi informaticienne, Moi mathématicienne**).
- ▶ Environnement bienveillant (réunions en **non-mixité** ou **mixité choisie**).
- ▶ Mettre en évidence des femmes avec un parcours inspirant.

## Relations avec l'industrie :

- ▶ Projet de recherche axé sur la vérification de systèmes temps réel.
- ▶ Liens potentiels entre mes recherches et les industries de systèmes critiques (**Airbus**).

## Approche pédagogique :

- ▶ Développement de logiciels de visualisation pour les enseignements théoriques.
  - ▶ Comparaisons de l'efficacité d'algorithmes pour **Initiation à l'Algorithmique**.  
Dès le premier semestre de la **première année**.
  - ▶ Outil **générique** pour visualisation des calculs d'automates pour **Calculabilité et complexité**.  
Dès le premier semestre de la **deuxième année**.
- ▶ Implémentations modulaires, prévues pour être facilement adaptées.

# Mes productions logicielles

## Productions scientifiques :

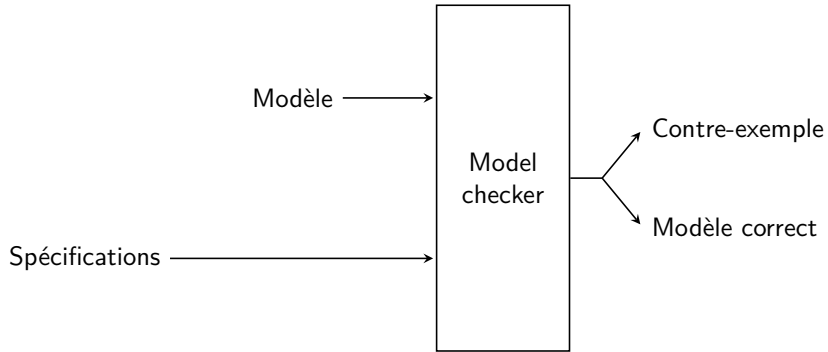
- ▶ Implémentation en **Python** pour [CIKM'20].
- ▶ Implémentations en **Java** pour [TACAS'22] et [TACAS'23].  
**Seul développeur.**
- ▶ Optimisation d'un code **Python** pour [SYNTCOMP].
- ▶ Implémentation en **C++** pour [Soumission MFCS].  
**Seul développeur.**  
Environ 5500 lignes de code en 1 mois.

## Productions non-scientifiques :

- ▶ Moteur de jeux vidéos en **C++**.
- ▶ Petits jeux vidéos en **C**.
- ▶ Mon site Internet en **Python** (nouvelle version en **Haskell** en cours d'écriture).
- ▶ Puzzles d'algorithmique, notamment avec **Haskell**.
- ▶ Concours de programmation en **Python** et **C++**.

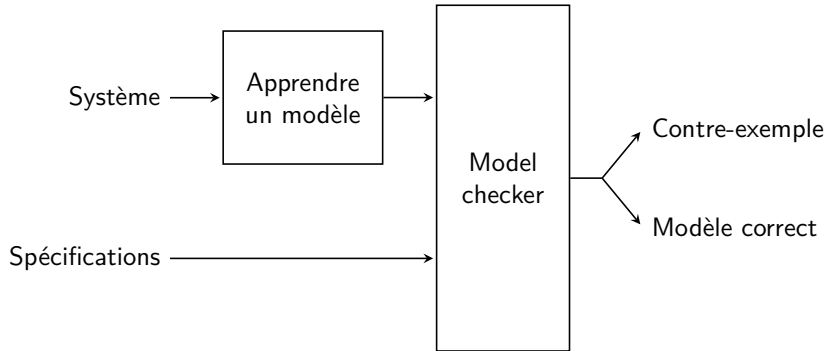
Maîtrise de **plusieurs langages de programmation**, dans des **paradigmes différents**.

# Projet de recherche – Vérification de modèles

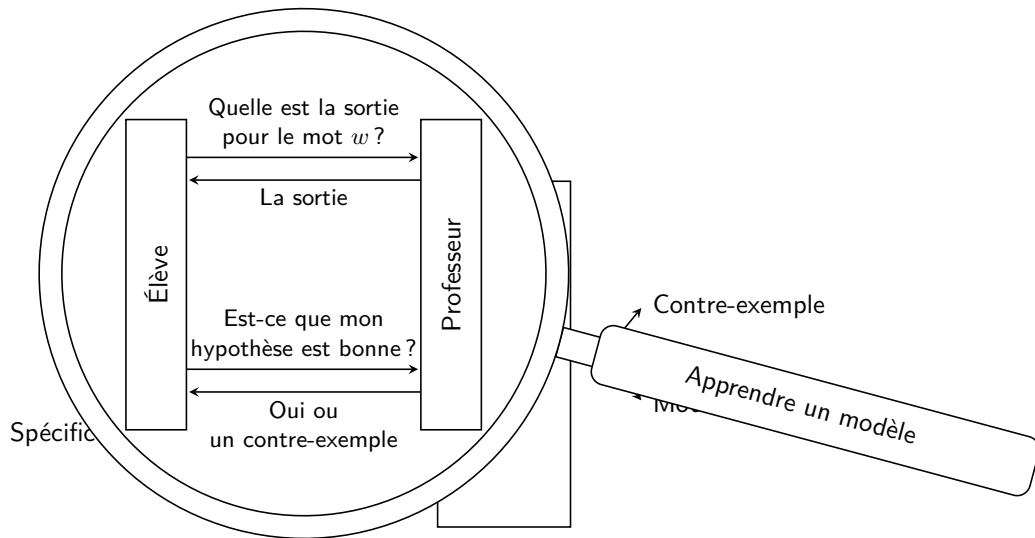




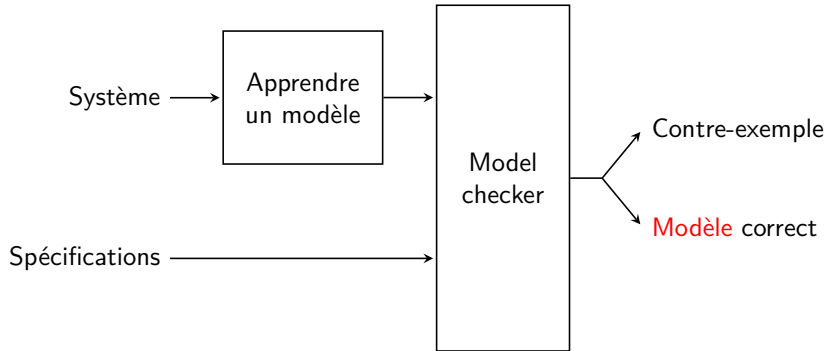
# Projet de recherche – Vérification de systèmes



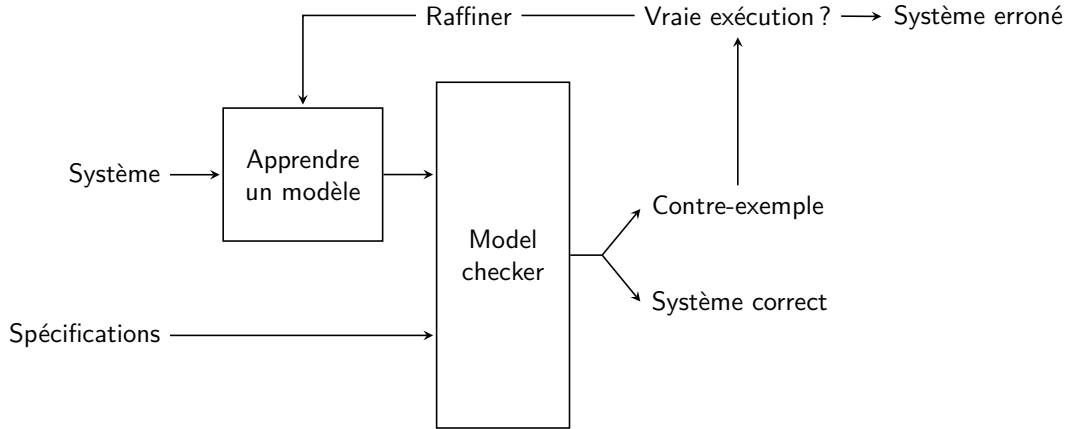
# Projet de recherche – Vérification de systèmes



# Projet de recherche – Vérification de systèmes

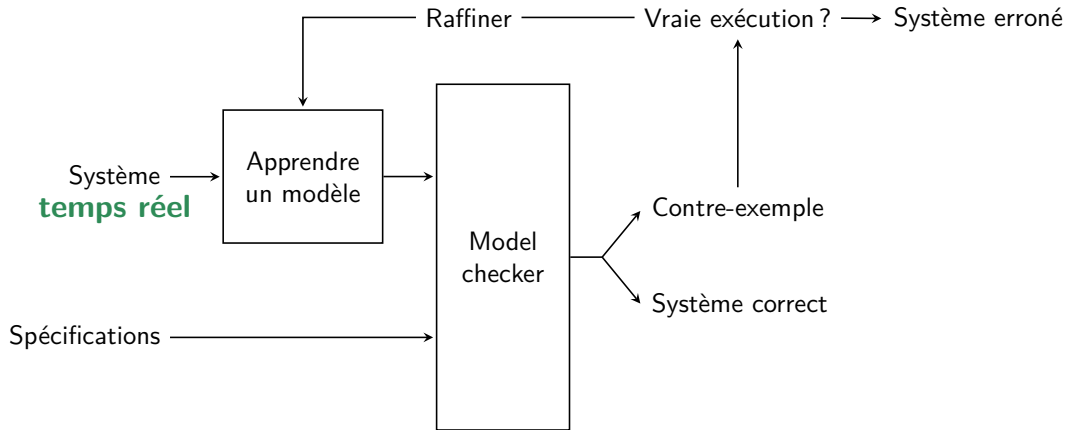


# Projet de recherche – Vérification de systèmes



Voir D. Dennis et O. Grumberg, *Abstraction and abstraction refinement*, Handbook of Model Checking, 2018.

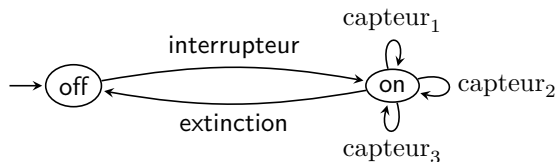
# Projet de recherche – Vérification de systèmes **temps réel**



# Exemple – automate

Système d'extinction automatique d'une lampe :

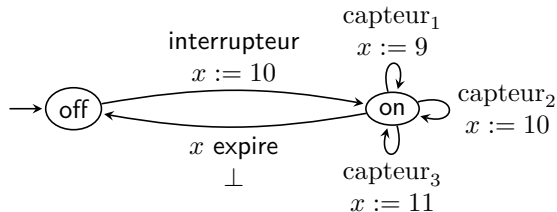
- ▶ 10 minutes après appui sur l'interrupteur ;
- ▶ si présence détectée, à nouveau 10 minutes ;
- ▶ bugs : un capteur met la contrainte de temps à 9 minutes, et un autre à 11 minutes.



## Exemple – automate **avec minuteurs**

Système d'extinction automatique d'une lampe :

- ▶ 10 minutes après appui sur l'interrupteur ;
- ▶ si présence détectée, à nouveau 10 minutes ;
- ▶ bugs : un capteur met la contrainte de temps à 9 minutes, et un autre à 11 minutes.



**Automates temporisés (horloges)**

**Automates à minuteurs**



# Horloges et minuteurs

## **Automates temporisés (horloges)**

- ▶ Horloges de 0 à l'infini ;
- ▶ Accès aux valeurs des horloges ;

## **Automates à minuteurs**

- ▶ Minuteurs d'une valeur donnée à 0 ;
- ▶ Seulement tests pour zéro ;

# Horloges et minuteurs

## Automates temporisés (horloges)

- ▶ Horloges de 0 à l'infini ;
- ▶ Accès aux valeurs des horloges ;
- ▶ Plus expressifs et bien connus ;

## Automates à minuteurs

- ▶ Minuteurs d'une valeur donnée à 0 ;
- ▶ Seulement tests pour zéro ;
- ▶ Plus restrictifs et moins connus ([[FORMATS'23](#)]) ;

# Horloges et minuteurs

## Automates temporisés (horloges)

- ▶ Horloges de 0 à l'infini ;
- ▶ Accès aux valeurs des horloges ;
- ▶ Plus expressifs et bien connus ;
- ▶ Apprentissage coûteux (Waga (2023). Active Learning of Deterministic Timed Automata with Myhill-Nerode Style Characterization) ;

## Automates à minuteurs

- ▶ Minuteurs d'une valeur donnée à 0 ;
- ▶ Seulement tests pour zéro ;
- ▶ Plus restrictifs et moins connus ([[FORMATS'23](#)]) ;
- ▶ Apprentissage plus facile ([[Soumission QEST+FORMATS](#)]) ;

# Horloges et minuteurs

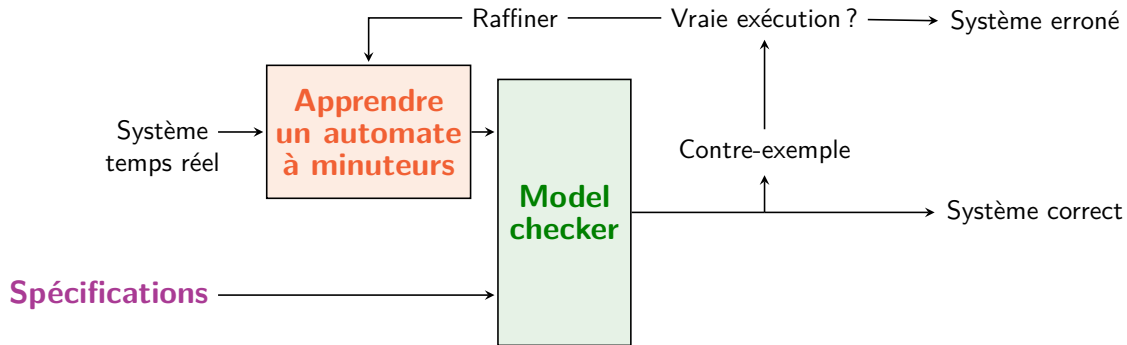
## Automates temporisés (horloges)

- ▶ Horloges de 0 à l'infini ;
- ▶ Accès aux valeurs des horloges ;
- ▶ Plus expressifs et bien connus ;
- ▶ Apprentissage coûteux (Waga (2023). Active Learning of Deterministic Timed Automata with Myhill-Nerode Style Characterization) ;
- ▶ Outils de model checking efficaces (UPPAAL, TChecker, IMITATOR).

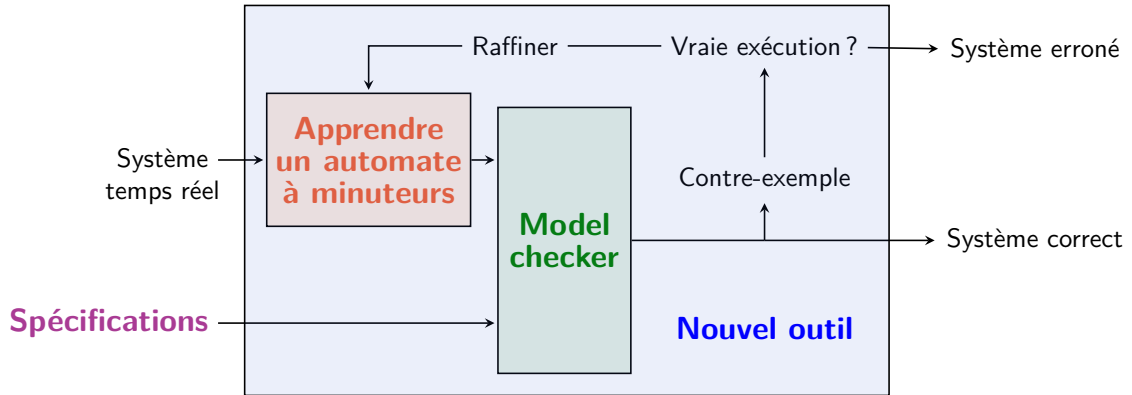
## Automates à minuteurs

- ▶ Minuteurs d'une valeur donnée à 0 ;
- ▶ Seulement tests pour zéro ;
- ▶ Plus restrictifs et moins connus ([[FORMATS'23](#)]) ;
- ▶ Apprentissage plus facile ([[Soumission QEST+FORMATS](#)]) ;
- ▶ Pas d'outils.

# Projet de recherche – Vue générale



# Projet de recherche – Vue générale



# Projet de recherche – Objectifs

## **Théorie des automates à minuteurs.**

- ▶ Équivalence avec une sous-famille des automates temporisés ?
- ▶ Algorithmes spécifiques et plus efficaces en pratique.

## **Spécifications.**

- ▶ Logiques existantes : TPTL, ECTL ?
- ▶ Nouvelle(s) logique(s).

## **Apprentissage.**

- ▶ Modèles plus expressifs.
- ▶ Apprentissage robuste aux variations dans la mesure du temps.
- ▶ Implémentations.

## **Vérification.**

- ▶ Étendre TChecker.

## **Model checking de systèmes temps réel : cas d'étude.**

- ▶ Collection de benchmarks.
- ▶ Apprentissage de systèmes :  
Protocoles réseaux ;                      Systèmes embarqués ;                      Systèmes d'exploitation temps-réel.
- ▶ Model checking.

# Intégration dans le département Méthodes et Modèles Formels

**Équipe MTV.** Vérification de systèmes temps réel.

**Équipe LX.** Théorie des automates.

**Équipe Synthèse.** Apprentissage d'automates.

## Collaborations internes.

- ▶ Frédéric Herbreteau.
- ▶ Igor Walukiewicz.
- ▶ Mathieu Hilaire.
- ▶ Anca Muscholl.
- ▶ Vincent Penelle.

## Collaborations externes.

- ▶ Nijmegen et Eindhoven (Pays-Bas) : Frits Vaandrager et Pieter Cuijpers.
- ▶ Mons et Anvers (Belgique) : Véronique Bruyère et Guillermo Pérez.
- ▶ Rennes et Paris-Saclay : Nathalie Bertrand et Patricia Bouyer.

## Projets auxquels je pourrais contribuer.

- ▶ ANR PaVeDys – Parametric Verification of Dynamic Distributed Systems.
- ▶ ANR UNREAL – UNderstanding Regular Expressions, Automata and Logics.



## Parcours académique :

2024 – Présent Postdoctorat à Rennes

Août – Oct. 2023 Séjour à Nijmegen.

2020 – 2024 Doctorat à Mons et Anvers.

## Activités collectives :

- ▶ Membre de plusieurs Conseils et Comités.
- ▶ Encadrement de 5 stagiaires.
- ▶ Comité évaluation artefact QEST+FORMATS 2025.
- ▶ Vulgarisation scientifique.

## Contributions :

- ▶ 4 conférences internationales, dont CIKM et TACAS (x2) ;
- ▶ 1 journal international : STTT ;
- ▶ 2 soumissions : MFCS, QEST+FORMATS ;
- ▶ Plusieurs réalisations logicielles.

## Enseignement :

- ▶ 251h équivalent TD.
- ▶ Capacité d'enseigner dès la rentrée 2025 :
  - ▶ Informatique théorique ;
  - ▶ Programmation et algorithmique ;
  - ▶ Intelligence artificielle ;
  - ▶ Mathématiques pour l'informatique.
- ▶ Responsabilités envisagées :
  - ▶ Encadrement d'élèves ;
  - ▶ Suivis de stage.
  - ▶ Implication dans la vie de l'ENSEIRB-MATMECA.
  - ▶ Féminisation et inclusion.
- ▶ Pédagogie via des outils visuels.

**Recherche :** Apprentissage et vérification de modèles pour des systèmes temps-réel.

**Intégration** Équipes MTV, LX et Synthèse.

**Collaborations** Nijmegen, Eindhoven, Mons, Anvers, Paris-Saclay, Rennes.