

FICHE DE POSTE

Apprenti chercheur – Apprentie chercheuse en Sciences du Numérique

| STRUCTURE D'ACCUEIL DE L'APPRENTI.E | |
|---|------------------------------------|
| Nom du laboratoire d'accueil | Laboratoire I3S |
| Nom de l'équipe de recherche d'accueil | COMRED |
| Prénom et Nom du maitre ou de la maitresse d'apprentissage | Cinzia DI GIUSTO |
| E-mail de contact du maitre ou de la maitresse d'apprentissage | Cinzia.di-giusto@univ-cotedazur.fr |
| Localisation géographique de l'équipe de recherche d'accueil et du bureau de l'apprenti.e | Sophia Antipolis, Bâtiment Euclide |

| ENCADREMENT DE L'APPRENTI.E |
|---|
| <p>L'étudiant viendra au laboratoire I3S les jeudis et vendredis et plus généralement toutes les demi-journées non enseignées. En particulier, il aura deux semaines complètes en immersion durant l'année (une par semestre), ainsi que les pauses pédagogiques et les vacances d'été.</p> <p>Le candidat devra avoir des connaissances en informatique fondamentale de niveau L3 informatique (théorie des automates, logique, algorithmique, fondements des langages de programmation, etc).</p> |

| SUJET DE RECHERCHE (1 page maximum) |
|---|
| <p>Gestion des défaillances dans les systèmes distribués</p> <p>Le fonctionnement des systèmes distribués modernes, tels que les architectures de microservices ou les plateformes cloud repose sur de nombreuses interactions entre composants indépendants, ce qui les rend particulièrement sensibles aux défaillances : pannes de machines, pertes de messages ou interruptions de communication. Ces incidents, même lorsqu'ils sont locaux, peuvent rapidement se propager et perturber l'ensemble du système. Dans ce contexte, ce sujet propose d'explorer comment les méthodes formelles peuvent aider à concevoir des systèmes capables de détecter, comprendre et corriger ces défaillances.</p> <p>L'objectif du stage est de développer une approche intégrée combinant plusieurs aspects complémentaires : la détection des erreurs à l'exécution, l'analyse de leur propagation, la mise en place de mécanismes de reprise (comme le retour à un état précédent) et l'adaptation dynamique du système. À travers ce sujet, l'étudiant sera amené à manipuler des modèles formels (les types</p> |

de session multiparty, les open automata, les automates communicantes) à comprendre leurs propriétés, et éventuellement à proposer des outils ou des prototypes illustrant ces concepts.

Les recherches actuelles proposent déjà quelques outils pour améliorer la fiabilité des communications [2,9]. Les types de session multiparty constituent aujourd'hui une approche reconnue pour spécifier et vérifier des protocoles distribués de manière rigoureuse. Ils garantissent notamment que les échanges entre composants respectent une structure bien définie et qu'aucun blocage ne survient lors de l'exécution. Par ailleurs, plusieurs extensions ont été proposées pour prendre en compte les défaillances [1,3,4,6,7]. Certaines introduisent des mécanismes d'exception permettant de réagir à des erreurs, d'autres permettent de définir des comportements alternatifs en cas de panne [5,8]. Des travaux ont également exploré l'utilisation du monitoring à l'exécution pour détecter des anomalies, ou encore des techniques de compensation et de rollback pour restaurer un état cohérent après une erreur. Cependant, ces approches restent souvent limitées à un aspect particulier du problème. Par exemple, la détection des erreurs est rarement étudiée conjointement avec leur propagation dans le système, et les mécanismes d'adaptation dynamique sont généralement traités séparément des garanties de correction formelle.

Le principal défi scientifique de ce sujet réside dans la difficulté à concilier plusieurs exigences souvent contradictoires. D'un côté, les systèmes distribués sont par nature complexes et évolutifs, ce qui rend leur modélisation délicate. De l'autre, les méthodes formelles exigent des modèles suffisamment structurés pour permettre des analyses rigoureuses. Un premier verrou concerne la compréhension de la propagation des défaillances. Lorsqu'un composant tombe en panne, il est difficile de déterminer précisément quels autres composants sont affectés et comment l'information sur la panne doit être partagée pour maintenir une cohérence globale. Un second enjeu important est l'intégration du monitoring avec les mécanismes de reprise. Il ne suffit pas de détecter une erreur : encore faut-il être capable de réagir de manière correcte, par exemple en revenant à un état antérieur sans introduire d'incohérences dans le système. Cette question est étroitement liée aux travaux sur la réversibilité dans les systèmes concurrents. Enfin, un défi majeur consiste à permettre l'adaptation dynamique des systèmes en présence de défaillances. Dans des architectures modernes, il est souvent nécessaire de reconfigurer les interactions entre composants, de remplacer un service défaillant ou de modifier le comportement global. L'enjeu est alors de garantir que ces adaptations préservent les propriétés de sûreté des communications et n'introduisent pas de nouveaux problèmes.

Références

1. Fabbretti, G., Lanese, I. et Stefani, J. A Behavioral Theory for Distributed Systems with Weak Recovery. *Log. Methods Comput. Sci.*, 21(, 3), 2025.
2. Glabbeek, R. van. Failure Trace Semantics for a Process Algebra with Time-outs. *Logical Methods in Computer Science*, 17(, 2), 2021, 11 :1-11 :40.
3. Harvey, P., Hu, R., Peters, K. et Yoshida, N. Multiparty Session Types for Safe Runtime Adaptation in an Actor Language. In *European Conference on Object-Oriented Programming (ECOOP) 2021(LIPIcs. 194)*, 2021,10 :1-10 :30.
4. Neykova, R. et Yoshida, N. Let It Recover : Multiparty Protocol-Induced Recovery. In *Compiler Construction (CC) 2017*, 2017, p. 98-108.
5. Di Giusto, C. et Pérez, J. A. Event-Based Run-Time Adaptation in Communication-Centric Systems. *Formal Aspects of Computing*, 28(, 4), 2016, p. 531-566.
6. Fowler, S., Hu, R., Yoshida, N. et Honda, K. Exceptional Asynchronous Session Types : Session Types without Tiers. In *POPL*, 2016, 28 :1-28 :29.
7. Honda, K., Yoshida, N. et Carbone, M. Multiparty Asynchronous Session Types. *J. ACM*, 63(1), 2016, 9 :1-9 :67.
8. Di Giusto, C. et Pérez, J. A. Disciplined structured communications with disciplined runtime adaptation. *Sci. Comput. Program.*, 97, 2015, p. 235-265.

9. Francalanza, A. et Hennessy, M. A Theory for Observational Fault Tolerance. Journal of Logic and Algebraic Methods in Programming, 73, 2007, p. 22-50.

ACTIVITES CONFIEES A L'APPRENTI.E

L'apprenti chercheur- apprentie chercheuse approfondira les savoirs et savoir-faire suivants :

- Comprendre le contexte de la recherche scientifique (système des publications, des conférences, fonctionnement par projet, etc), méthodologie de la réflexion scientifique (conjectures, preuves, recherches d'exemples et de contre-exemples, etc) et de la production et communication scientifique (dépôt ouvert de code, utilisation de latex, travail bibliographique, exposés oraux, etc).
- Acquérir des connaissances sur le sujet scientifique, analyser l'état de l'art, compiler une bibliographie, publier des résultats scientifiques, faire face à l'évaluation du travail scientifique par des pairs et présenter l'activité scientifique à un public spécialisé.
- Planification, réalisation et analyse d'expériences scientifiques.
- Acquérir des connaissances sur la modélisation formelle : Algèbre de processus, systèmes à automates, Systèmes de typage
- Acquérir des connaissances sur la mécanisation de preuve à travers le démonstrateur Rocq
- Implémentation d'outil pour la reconnaissance de failles

COMPETENCES REQUISES POUR REALISER LES ACTIVITES

Le candidat devra avoir des connaissances en informatique fondamentale de niveau L3 informatique (théorie des automates, logique, algorithmique, fondements des langages de programmation, etc).

PERSPECTIVES APRES LA PERIODE D'APPRENTISSAGE

Ce poste d'apprenti chercheur – apprentie chercheuse a vocation à préparer le candidat ou la candidate à une éventuelle poursuite en doctorat.

Le travail de recherche qui sera mené pendant le contrat d'apprentissage pourra donner lieu à des publications scientifiques, où l'apprenti.e pourra être co-auteur dans la perspective d'une thèse.