The initial contract is for one year, but solid progress on research goals would open other international workshops, conferences, and journals.

Models, proposing verification techniques, implementing plugins and tactics in the Coq ITP, and the tools for interactively verifying Lustre programs in the context of the Vélus project. Please see verification goals, safely mixing results from model-checkers and specialized tactics, and transferring model-based development. We believe that ITPs provide an ideal environment for decomposing formal semantics for block-diagram specifications. Lustre is a synchronous language closely related to the industrial Scade tool which supports Model-Based Development (MBD) of critical embedded control software. It's a programming language with a formal semantics for block-diagram specifications.

Vélus is developed in the Inria PARKAS team located at the École normale supérieure in the heart of Paris. Development is led by Timothy Bourke and involves collaboration with experts in synchronous languages (Marc Pouzet), verified compilation (Xavier Leroy) and automatic verification (Cesare Tinelli).

Work on Vélus has to date focused on verified compilation. This postdoctoral position would involve developing techniques and tools for interactively verifying Lustre programs within Coq. That is, making it possible to load a Lustre program into Coq, to state invariants, and to prove that the program satisfies them. The compiler correctness theorem guarantees that such invariants also hold making it possible to load a Lustre program into Coq, to state invariants, and to prove that the generated assembly code.

Vélus is developed in the Inria PARKAS team located at the École normale supérieure in the heart of Paris. Development is led by Timothy Bourke and involves collaboration with experts in synchronous languages (Marc Pouzet), verified compilation (Xavier Leroy) and automatic verification (Cesare Tinelli).

Mission confiée

Work on Vélus has to date focused on verified compilation. This postdoctoral position would involve developing techniques and tools for interactively verifying Lustre programs within Coq. That is, making it possible to load a Lustre program into Coq, to state invariants, and to prove that the program satisfies them. The compiler correctness theorem guarantees that such invariants also hold making it possible to load a Lustre program into Coq, to state invariants, and to prove that the generated assembly code.

Work on Vélus has to date focused on verified compilation. This postdoctoral position would involve developing techniques and tools for interactively verifying Lustre programs within Coq. That is, making it possible to load a Lustre program into Coq, to state invariants, and to prove that the program satisfies them. The compiler correctness theorem guarantees that such invariants also hold making it possible to load a Lustre program into Coq, to state invariants, and to prove that the generated assembly code.

The Vélus compiler is developed in OCaml and the Coq Interactive Theorem Prover (ITP). It successively transforms source programs into the C light language used by CompCert. The overall correctness theorem states that the stream-based semantics of the source program is preserved by the generated assembly code.

Vélus is a synchronous language closely related to the industrial Scade tool which supports Model-Based Development (MBD) of critical embedded control software. It's a programming language with a formal semantics for block-diagram specifications.

Lustre is a synchronous language closely related to the industrial Scade tool which supports Model-Based Development (MBD) of critical embedded control software. It's a programming language with a formal semantics for block-diagram specifications.

The Vélus compiler is developed in OCaml and the Coq Interactive Theorem Prover (ITP). It successively transforms source programs into the C light language used by CompCert. The overall correctness theorem states that the stream-based semantics of the source program is preserved by the generated assembly code.

Work on Vélus has to date focused on verified compilation. This postdoctoral position would involve developing techniques and tools for interactively verifying Lustre programs within Coq. That is, making it possible to load a Lustre program into Coq, to state invariants, and to prove that the program satisfies them. The compiler correctness theorem guarantees that such invariants also hold making it possible to load a Lustre program into Coq, to state invariants, and to prove that the generated assembly code.

The postdoctoral researcher will collaborate closely with other team members to develop techniques and tools for interactively verifying Lustre programs in the context of the Vélus project. Please see the Vélus website for scientific references. This work will involve mastering the existing semantic models, proposing verification techniques, implementing plugins and tactics in the Coq ITP and working on case-studies to guide and evaluate the approach. Results will be communicated to international workshops, conferences, and journals.

Initial contract is for one year, but solid progress on research goals would open other opportunities.

1. Master relevant background knowledge on synchronous languages and the state-of-the-art in interactive program verification by reading books and research articles.
2. Experiment with small examples in the Coq ITP to understand the problem and potential solutions.
3. Propose verification techniques and implement them as Coq plugins and tactics.
4. Evaluate the techniques and implementations on one or more larger case studies.
5. Document the results in technical reports and scientific articles.

Additional activities:
1. Participate in workshops and team activities.
2. (Optional) Teach at the L3 and masters level.

Compétences

Technical skills and level required:
- Strong expertise in one or more programming languages (ideally a strongly typed functional language like OCaml, SML, or Haskell).
- Some experience with Interactive Theorem Proving or a very strong motivation to become a specialist.

Languages: French or English.

Avantages

- Subsidized meals
- Partial reimbursement of public transport costs
- Leave: 7 weeks of annual leave + 10 extra days off due to RTT (statutory reduction in working hours) + possibility of exceptional leave (sick children, moving home, etc.)
- Possibility of teleworking (after 6 months of employment) and flexible organization of working hours
- Professional equipment available (videoconferencing, loan of computer equipment, etc.)
- Social, cultural and sports events and activities
- Access to vocational training

Informations générales

- Thème/Domaine : Preuves et vérification
- Ville : Paris
- Centre Inria : CRI de Paris
- Date de prise de fonction souhaitée : 2020-01-01
- Durée de contrat : 1 an
- Date limite pour postuler : 2020-02-29

Contacts

- Equipe Inria : PARKAS
- Recruteur : Bourke Timothy / Timothy.Bourke@inria.fr

A propos d’Inria

Inria, l’institut national de recherche dédié aux sciences du numérique, promeut l’excellence scientifique et le transfert pour avoir le plus grand impact. Il emploie 2400 personnes. Ses 200 équipes-projets agiles, en général communes avec des partenaires académiques, impliquent plus de 3000 scientifiques pour relever les défis des sciences informatiques et mathématiques, souvent à l’interface d’autres disciplines. Inria travaille avec de nombreuses entreprises et a accompagné la création de plus de 160 start-up. L’institut s’efforce ainsi de répondre aux enjeux de la transformation numérique de la science, de la société et de l’économie.

L’essentiel pour réussir

You are passionate (but not fanatical!) about functional languages and interactive proof assistants. You are not just interested in the theory of logic, but really motivated to apply it to real-world problems. Tenacity and a taste for simplicity are essential character traits for this work.

You have completed, or will soon complete, a PhD in computer science, computer engineering, or mathematics with a strong emphasis on formal methods, verification, programming language semantics, or computer-assisted logic. You already have some experience with interactive theorem proving, but may not (yet) be an expert. Ideally, you have already contributed to software projects or publications.

Knowledge of Lustre, Scade, or Model-Based Development is not essential.

Consignes pour postuler

Sécurité défense :
Ce poste est susceptible d’être affecté dans une zone à régime restrictif (ZRR), telle que définie dans le décret n°2011-1425 relatif à la protection du potentiel scientifique et technique de la nation (PPST). L’autorisation d’accès à une zone est délivrée par le chef d’établissement, après avis ministériel favorable, tel que défini dans l’arrêté du 03 juillet 2012, relatif à la PPST. Un avis ministériel défavorable pour un poste affecté dans une ZRR aurait pour conséquence l’annulation du recrutement.

Politique de recrutement :
Dans le cadre de sa politique diversité, tous les postes Inria sont accessibles aux personnes en situation de handicap.

Attention: Les candidatures doivent être déposées en ligne sur le site Inria. Le traitement des candidatures adressées par d'autres canaux n'est pas garanti.