Date: / /

(month) (day) (year)

Application Form for Doctoral Research Fellow (DRF)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Vu Xuan Tung | Gender | □ Male □ Female |
| Intended Faculty Advisor | Prof. Mizuhito Ogawa | Date of Birth | / / |
| (month) (day) (year) |
| Intended School | □ School of Knowledge Science □ School of Information Science  □ School of Materials Science | | |
| Applying for | □ October 2014 □ April 2015 | | |

|  |  |
| --- | --- |
| \*Title of Intended Research During Fellowship | Unsatisfiability proof generation and interpolation for polynomial constraints over the reals |

\*Please consult the intended faculty advisor before filling in.

【Employment Record at JAIST】(only for those who have been enrolled in JAIST)

|  |  |
| --- | --- |
| Period of Employment | Position and Type of Work |
| From / / To / /  　 (month) (day) (year) (month) (day) (year) |  |
| From / / To / /  　 (month) (day) (year) (month) (day) (year) |  |
| From / / To / /  　 (month) (day) (year) (month) (day) (year) |  |
| From / / To / /  　 (month) (day) (year) (month) (day) (year) |  |
| From / / To / /  　 (month) (day) (year) (month) (day) (year) |  |

**Please note the following when you fill out the form.**

* The font size must be at least 10 points.
* Fill out the application form either in Japanese or in English.
* Do not alter the form in any way (i.e. do not change the size of the text boxes, do not add any new items, do not remove any items even if they are not applicable to you).
* No additional papers should be appended to the prescribed application form.
* This application will not affect the admission decision. Describe positively the ingenious and creative theme that you wish to work on.
* The application form must be printed double-sided.

――――――――――――――――――――――――――――――――――――――――――――――――

1. **Research Status** (Charts may be used to describe it clearly. Do not alter the form in any way. (The same applies hereinafter.))
2. Describe the background of the research, the problems, the solutions, the research purpose, the research procedures, the characteristics, and the ingenious points citing important literature in the field.
3. Explain the research progress and results obtained to date relating to the items described in 1) including the problems. If the research results have been published in a treatise or presented at an academic conference, clarify the section on which the applicant worked and describe them.

|  |
| --- |
| Satisfiability Modulo Theories (SMT) problem is a decision problem for logical formulas with respect to background theories capturing the meaning of the formulas. There is a number of background theories such as theory of real numbers, theory of integers, and the theories of various data structures such as lists, arrays and bit vectors. Based on the background theories, SMT solvers decide whether a set of formulas is satisfiable or not. If it is (SAT), a satisfying assignment of variables is returned; otherwise (UNSAT) the proof of un-satisfiability is optionally generated. SMT has a wide range of applications, namely test-case generation and model checking. We have developed an SMT solver called raSAT [1], which solves polynomial constraints overs reals (logic division of quantifier-free non-linear reals arithmetic, QF\_NRA). We choose QF\_NRA because of the following two reasons:   1. In SMT-LIB, there are not yet defacto standard methodologies for benchmarks on QF\_NRA. The candidates are:    * Quantifier elimination by cylindrical algebraic decomposition (QE-CAD) was implemented in QEPCAD-B, Mathematica, Reduce/Redlog. DEXPTIME complexity in the number of variables is the problem of QE-CAD.    * Interval Arithmetic is applied in many tools such as RSOLVER, dReal and iSAT.    * Bit-blasting: Input formulas are bit-blasting to propositional formulas which are then solved by SAT solver. MiniSmt and UCLID applied this method.    * Linearization is often used over integers. For example, Barcelogic linearizes constraints by instantiating an argument of a multiplication with all possible integers in a given bound. CORD uses another linearization for real numbers.    * Virtual substitution is applied in Z3, Mathematica and Reduce. 2. Many applications are encoded into polynomial constraints such as:    * Automated detection of round-off and overflow error [2] [3] which is the initial motivation of our work.    * Automatic termination proving of term rewriting system: This is a second order SAT problem which is undecidable. It can be solve by interpreting terms into polynomials over the reals and amounting termination conditions to the constraints over the coefficients of such polynomials.    * Invariant generation: [4] proposed a method of generating linear invariant using non-linear constraints solving based on Farkas’s Lemma.   raSAT focuses on inequality since inequality allows approximations but equality does not. raSAT uses raSATloop, a method of iterative approximation refinement, to solve polynomial constraints. Interval arithmetic (IA) (over-approximation) is used for disproving the constraints and testing (under-approximation) is used for proving. When raSAT neither proves nor disproves the constraints, refinements are applied to decompose the intervals of variables. Following is the problems which are supposed to be solved by my current research:   1. **SAT, UNSAT confirmation**: Roundoff, overflow errors can make our SAT and UNSAT result unsound. We plan to use iRRAM, package for error-bounded real arithmetic, to verify raSAT results. 2. **UNSAT core computation:** By making statistics on running time of raSAT, we detected that UNSAT core takes quite long time relatively to the total running time. 3. **Testing phase:** Statistics also showed that testing phase also accounts for a long running time. We plan to reduce the number of testing cases by focusing on only sensitive variables when generating test cases. 4. **Equality handling:** We will the intermediate value theorem first to simply solve the equalities.   With the above problems, the current status of my research is as follow.   1. Our work is going to be presented at SMT Workshop 2014. raSAT also participated the SMT competition 2014 on QF\_NRA. Our experiments on the competition’s servers show that raSAT solved 57 problems of zankl family which is beyond the winner of 2010 competition (MiniSmt) 2. **SAT, UNSAT confirmation**: I implemented SAT verification using iRRAM. UNSAT case is left for future work. 3. **UNSAT core computation**: Because UNSAT requires exhaustive search for all the possibilities, we define UNSAT core to reduce the target constraints. Possible choices are:    * **A subset of constraints from the original ones** : It is clear that if is UNSAT then is also UNSAT. We are investigating how to choose such subset. In the current research, I sorted the APIs using dependency between polynomials. For example, if all variables of f are also variables of g.    * **A sub-polynomials of in an API** : such that UNSAT of results in UNSAT of in a box. This idea was implemented in the previous work [1].    * **A subset of variables appearing in an API**  such that in a box are enough to lead UNSAT. This idea is new and was implemented in the current research. 4. **Testing phase**: If for each variable we generate 2 test cases, totally we will have test cases with n is the number of variables. Testing phase will consume much time if becomes large. We have the following solutions:  * Computation of affine interval maintains coefficient of the noise error reflecting the influence of a variable on the value of the polynomial. I implemented the strategy of prioritizing most influential variables in testing.   + When the intervals of variables are the same, we will probe on direction such as . Then, we can use the Sturm sequence as heuristics on test data generation. This is our future work in Master course. |

Name Vu Xuan Tung

|  |
| --- |
| * Also with the idea of Sturm sequence, we consider the multivariate polynomial, for instance, f(x, y) as a univariate polynomial g(x) with intervals as coefficients by IA as over-approximation. This is our future work in Master course.   + I also improve the implementation of testing phase reduce the running time and memory consumption.  1. **Equality handling**: We consider a simple approach of using the intermediate value theorem. First, we find a box of variables in which all inequality constraints are valid.   For single equation, suppose it is . We will find two points and inside the box such that \*. Because is continuous (it is a polynomial), we can conclude that the equation has one solution lying between and . Since all the in-equations are valid, the whole problem is SAT. In current research, we added handling for single equation.  For multiple equations, suppose we have two equations and and the set of variables in and is . Let the current considered box is and . If, for example:   * + for some and each and,   + for some and each   We can conclude that there exist a common solution of and inside the box. Therefore the whole problem is SAT. We are implementing the case of multiple equations.   1. **NIA implementation**: We will also extend raSAT for nonlinear integer arithmetics. One possible way is setting the minimum searching box to 1.  References 1. **To, Van Khanh and Ogawa, Mizuhito.** *raSAT: SMT for Polynomial Inequality.* s.l. : JAIST, 2013.  2. *Checking Roundoff Errors Using Counterexample-guided Narrowing.* **Ngoc, Do Thi Bich and Ogawa, Mizuhito.** Antwerp, Belgium : ACM, 2010. 978-1-4503-0116-9.  3. *Overflow and Roundoff Error Analysis via Model Checking.* **Ngoc, Do Thi Bich and Ogawa, Mizuhito.** s.l. : IEEE Computer Society, 2009. 978-0-7695-3870-9.  4. *Linear Invariant Generation Using Non-Linear.* **Michael A. Col´on, Sriram Sankaranarayanan and Henny B. Sipma.** |

1. **Research Plan**

**(1) Research Background**

Based on the research status in 1, describe the background of the research plan, the problems, the issues to resolve, how the idea occurred, etc., citing any references.

|  |
| --- |
| Up to here, there are two main problems to be considered for raSAT.   1. **Proof of UNSAT:** Unsatisfiability certificate can be used to extract interpolants which are widely known to be useful in invariant generation [5]. In addition, most of the current works focus on Presburger Arithmetic and there is not much research on interpolants of polynomial constraints. Interpolants of polynomial constraints arise during verification of complex systems such as hybrid ones.   C:\Users\tungvx\Desktop\proof.pngThe state-of-the-art SAT solvers has been improved quite much with combination of clause learning and restarts. They are able to generate good resolution proof (in terms of the size of the proof tree) when the UNSAT witness is found. MATHSAT [6] is an SMT solver supporting interpolation over the Linear Arithmetic. Like raSAT, it uses a DPLL-based SAT solver as the backend in the lazy manner. If, for example, SAT solver return a satisfiable solution and MATHSAT detects this solution as an inconsistency with respect to theory of arithmetic. The negation of that solution (maybe some UNSAT core) is learnt to SAT solver. If the new SAT formula is detected as UNSAT, resolution proof will be generated. Combining with the proof of the learnt clause, MATHSAT generates the desired interpolant. For instance, suppose we have the following two constraints: and and wee need to solve . First, SAT solver return as a solution. Then, MATHSAT detect an inconsistency. is added into SAT solver. Here, SAT solver will detect UNSAT and thus create the resolution proof as in Figure 1. Next, MATSAT generates proof for UNSAT of and calculates interpolant of A and B [6]. CSIsat also supports interpolation for LA and EUF.  Figure : Resolution example   1. **Grobner basis** for solving polynomial equalities. We apply the intermediate theorem for multiple equalities only when the numbers of equations must be smaller than or equal to the number of variables appearing in these equations. For complete equality solving over reals, Grobner basis is needed. Further, Grobner basis has been implemented in Maple and Mathematica, but the performance is quite limited. We expect to efficiently implement this technique and integrate it into raSAT.  References 5. *Abstractions from Proofs.* **Henzinger, Thomas A. and Jhala, Ranjit and Majumdar, Rupak and McMillan, Kenneth L.** Venice : ACM, 2004.  6. *Efﬁcient Generation of Craig Interpolants in Satisﬁability Modulo Theories.* **Alessandro Cimatti, Alberto Griggio, Roberto Sebastiani.** |

Name Vu Xuan Tung

**(2) Research Purpose and Content** (Charts may be used to describe them clearly.)

1. Describe the research purpose, the research procedures, and the research content.
2. Describe the plan in detail what and how far the applicant intends to discover.
3. If it is a collaborative research, specify the part that the applicant is in charge of.
4. If the applicant plans to carry out a research at another research institute (including overseas institutes, etc.) during the period of the research plan, describe the plan.

|  |
| --- |
| As mentioned, there are two main goals in this research proposal.   1. **UNSAT proof generation**.   Our primary idea will be illustrated by the following example though it is not so clear. Suppose we have the constraints formula: with the intervals: . By these initial intervals, IA cannot detect any conflicts. Therefore, decomposition is applied; the intervals become, for example, .  Suppose in the next iteration, the following intervals are selected: . At this point, raSAT detects that is inconsistent. Furthermore, is detected as the UNSAT core of . The derivation in Figure 2 can be generated because can only be in either or .  One more time, raSAT can detect a conflict here by considering: . Because and , becomes UNSAT. The proof tree is drawn as in Figure 2. From this proof, is extracted as the interpolant of and within selected intervals.  C:\Users\tungvx\Desktop\solution2.pngC:\Users\tungvx\Desktop\solution.png  Figure 2: Proof example 1  Figure 3: Proof example 2  Suppose we have two formulas and such that is UNSAT. We need to extract the interpolant of and . The intuition here is that we try to remove the intervals which make UNSAT, keeping the remaining intervals as the result of the derivation. Suppose the remaining intervals of after the first derivation is . Similarly, the remaining ones of is . We try to merge and The process of merging continue until conflicts are detected. At this point, the proof of UNSAT is completed. and may not necessary to be intervals, but some kinds of polynomial constraints such that and .   1. **Grobner basis**.   Generation of Grobner basis can be done by the approach of term rewriting. The computation is similar to the Knuth-Bendix completion procedures. The differences are:   * Grobner basis computation use real arithmetic. For example can be reduced to 0. * In order to reduce polynomials, associativity and commutativity of real arithmetic need to be used.   We will use Maude, a high performance rewriting language, to efficiently reduce polynomials using associative and commutative rewriting rules. Our group already implemented a completion tool called Maxcomp. We will investigate to see if we can extend it to handle Grobner basis computation.  **C:\Users\tungvx\Desktop\rewrite.png**The computation process starts by converting the set of equalities to rewriting rules over the polynomials (left hand side of each rule is always a monomial) with respect to a total order on power products (power product is a monomial without coefficients). The remaining process of building Grobner basis is similar to completion of term rewriting system. For example, suppose we have the set of equalities: . Suppose a total ordering on monomials results: , and . We have can convert F to the following rewriting system: . Next, we compute the critical pairs of R:  , and cannot be reduced any more. We first substract them, resulting a polynomial of . The reduced polynomial is converted to a rewriting rule: . This rule is then added into R.  The process continues with critical pairs computation, adding new rewriting rules until all the detected critical pairs are processed and no more new critical pairs are generated.   1. **Collaboration.**   Our work will be in collaboration with the works of other members in our research group.   * raSAT development is the collaboration between our lab and Dr. Khanh from University of Engineering and Technology, Vietnam. The development is mainly on our side. * The system of Binh and the team from Ho Chi Minh University of Technology will use raSAT to solve polynomials. |

Name Vu Xuan Tung

**(3) Characteristics and Ingenious Points of Research**

Describe the following:

1. If there are any earlier research, describe the characteristics, the focal point, and the ingenious point of the proposed research compared with them.
2. The position and meaning of the research among the related research.
3. The expected impact and outlook for the future when the research is completed.

|  |
| --- |
| 1. **raSATloop** applies intervals arithmetic and testing, which refine each other. If interval arithmetic decides the constraints as neither UNSAT nor IA-VALID (SAT for all possible values of variables in their ranges), testing will be implemented to find a SAT instance. iSAT, on the other hand, only applies interval arithmetic and has no U.T such as testing as in raSAT. As the result, it often fails to conclude SAT. In addition, while raSATloop combines both affine interval and classical interval, iSAT use classical interval only. dREAL does not conclude SAT but delta-sat which is generally does not imply SAT. 2. **UNSAT proof for QF\_NRA, QF\_NIA**. To our knowledge, there is little works on interpolants of polynomial constraints using UNSAT proof. CSIsat and MATHSAT supports interpolation for linear arithmetic. Nonlinear arithmetic is generally more expressive than linear arithmetic, and thus is able to be used in verifying more complicated system. 3. We believe that raSATloop with interval arithmetic and testing will become the standard method not only for QF\_NRA but also for QF\_NIA which is mostly implemented with bit-encoding with SAT solver. |

**(4) Annual Research Plan**

The allocation of the space for the years may be changed as long as it fits within the original textbox.

|  |
| --- |
| (1st year)  ＤＣ   1. We first attempt to implement calculation of Grobner basis from a set of equations. We will investigate the current completion tool Maxcomp to see if it can be used in our project. This phase should be finished within 3 months, from April to June. 2. We will also participate the SMT competition. In SMT competition 2014, we didn’t support all the syntax of SMT LIB. In addition, we also proposal a new families for QF\_NRA and submit to SMT LIB. 3. Optimizing the above implementation is needed to improve the performance of Grobner basis computation. Optimization will be implemented within 2 months. 4. After that, we need to integrate Grobner basis computation into raSAT. From the basis, we get the new set of equalities. We will solve these equalities. Currently raSAT use C language as the main. Interfacing C and Maude must be investigated. The estimated time for this task is 2 months. 5. We spend the remaining time of this year for finding the way how to formalize the UNSAT proof. Initial implementation should also be done to test the possibility.   (2nd year)   1. We continue generating UNSAT proof for polynomial constraints. Expected time is 6 months. 2. We will read the resolution proof of SAT solver, integrating proof of theory solver to form the complete proof. This should be done within 6 months.   (3rd year)   1. After having the formal proof of UNSAT, we will generate interpolants of polynomial constraints. This work should be done within 3 months. 2. We will do experiments on model checking hybrids system to see how our approach performs. This will be done until the end of this year. 3. We will write thesis in the last 9 months. |

Name Vu Xuan Tung

1. **Research Achievements** (List only the achievements in which the applicant played a major role for the following categories. Number them serially and write “N/A” for inapplicable items. Underline the applicant. If there are too many achievements to list, list only main achievements and write the number of the unlisted achievements (e.g. “and 3 other presentations”) at the end of each item.)

**(1) Books and published articles in scholarly journals, bulletins, miscellanies, etc.** (List whether it has been peer reviewed or not. Peer reviewed articles must have already been published or decided to be published. Exclude the articles currently undergoing peer review or in the application process.)

1. List Authors (names of all the authors including the applicant in the same order as the book or the article), Title, Name of Magazine in which the article is published, Publisher, Volume Number, pp [start page]–[end page], Year in this order. List the affiliations and positions of the authors in footnotes.
2. Submit proof for the articles that have been decided to be published.

**(2) Expositions and reviews in scholarly journals and commercial magazines**

**(3) Presentations at international academic conferences** (List whether it is verbal or poster, and peer reviewed or not peer reviewed)

(Exclude the ones yet to be presented. However, the ones for which applications have already been accepted may be listed. In that case documents to prove them need to be submitted.) List Authors (names of all the authors including the applicant in the same order as the article, etc.), title, name of conference, number of the article, location, month, and year. Circle the speaker.

**(4) Presentations at domestic academic conferences, symposiums, etc.**

Describe in the same way as in (3). If you list the ones for which applications have already been accepted, proof needs to be submitted in the same way as in (3).

**(5) Patents, etc.** (List whether it has been applied, published, or granted. If it is in the application process and details cannot be disclosed, list a brief summary.)

**(6) Other** (awards, etc.)

|  |
| --- |
| **(1)Books and published articles in scholarly journals, bulletins, miscellanies, etc.**  Vu, Xuan-Tung1 and Truong, Anh-Hoang2 and Tran, Mai Thuong3 and Steffen, Martin4, A Type System for Finding Upper Resource Bounds of Multi-threaded Programs with Nested Transactions, Proceedings of the Third Symposium on Information and Communication Technology, ACM, N/A, pp 21-30, 2012.  1. Student at VNU University of Engineering and Technology, Vietnam  2. PhD, Head of Software Engineering Department, VNU University of Engineering and Technology, Vietnam  3. PhD candidate at Department of Informatics, University of Oslo, Norway  4. Professor in computer science at Department of Informatics, University of Oslo, Norway  **(3) Presentations at international academic conferences**  (verbal presentation, not peer reviewed)  To Van Khanh, Vu Xuan Tung and Mizuhito Ogawa, raSAT: SMT for Polynomial Inequality, 12th edition of the International Worshop on Satisﬁability Modulo Theories (SMT 2014), N/A, Vienna, 7, 2014  **(6) Other**  Participated in SMT competition 2014 on QF\_NRA division. |

Name Vu Xuan Tung

**3. Research Achievements Sample**

This is only a sample. Adjust the format as needed in accordance with the instructions in the form.

|  |
| --- |
| **(1)Books and published articles in scholarly journals, bulletins, miscellanies, etc.**  (peer reviewed)  1) Taro Sentan1, Hanako Ishikawa2, ･･･”(Title)”, ‘(Magazine Name)’, ○○Publishing, Volume ○, pp57-62, 2003  2) Jiro Nomi3, Taro Sentan1, ･･･”(Title)”, ‘(Magazine Name)’, ○○Publishing, Volume○, pp33-39, 2009  and 5 other papers  List the number of unlisted papers or books for each category.  Note: Affiliations and Positions of Authors (as of the publish date)  1. Research Student at ○○ Department, ○○ University, 2. Assistant Professor at ○○ Department, ○○ University, 3. Professor at ○○Department, ○○ University, ･･･  List the affiliations and the positions of the authors.  **(2)Expositions and Reviews in Scholarly Journals and Commercial Magazines**  1) Taro Sentan･･･ “(Title)”, ‘(Magazine Name)’, ○○Publishing, Volume ○, pp57-62, 2006  **(3) Presentations at International Academic Conferences**  (verbal presentation, peer reviewed)  1) ○ Sentan T, Ishikawa H, ･･･ “(Title)”, ‘(Conference Name)’, BB-11, Los Angeles, USA, (June 2005)  and other 2 presentations  If it was numbered when published, include the number.  **(4)Presentations at Domestic Academic Conferences, Symposium, etc.**  (verbal presentation, not peer reviewed)  1) ○Taro Sentan, Hanako Ishikawa,･･･ “(Title)”, ‘(Conference Name)’, No.200, Sendai, September 2004  **(5)Patents**  (Published)  1) (Patent Number), ”(Name)”, Jiro Nomi, Taro Sentan, April 2004  **(6)Other (awards, etc.)**  1) Taro Sentan･･･’(Award Name)’, April 2012  Separate the past ones and the ones to be published/presented that needs proof documents.  **【To be published or presented】**  **(1)**Articles Decided to be published in Scholarly Journals, Bulletins, Miscellanies, etc.  (peer reviewed)  1) Jiro Nomi1, Taro Sentan2,･･･”(Title)”, ‘(Magazine Name)’, ○○Publishing, Volume ○, pp33-39, 2012  (proof 1 attached)  Note: Affiliations and Positions of Authors  1. Assistant Professor at ○○ Department, ○○ University, 2. Research Student at ○○ Department, ○○ University, ･･･  **(4)Presentations at domestic conferences or symposiums for which applications have been accepted**  1)○Taro Sentan, Hanako Ishikawa, ･･･”(Title)”, ’(Conference Name)’, No.201, Tokyo, July 2012 (proof 2 attached)  Proof documents may be a print out of e-mail, etc. Do not attach a research paper. Put the number (e.g. “Proof 1”) in the upper right corner of the document. |