

# Automatic test generation for functional software

ENSTA ParisTech + Naval Group



**NICOLAS BAUDIN**  
INTERNSHIPS IN FRANCE INITIATIVE

<b>Name of the hosting institution in France</b>	ENSTA ParisTech
<b>Name of the host laboratory / research team</b>	Computer Science and System Engineering Department
<b>Address</b>	828 bd des maréchaux 91120 Palaiseau, France
<b>Web site</b>	<a href="http://u2is.ensta-paristech.fr">http://u2is.ensta-paristech.fr</a>
<b>Name of the supervisor</b>	Alexandre Chapoutot
<b>Function</b>	Associate professor
<b>Email</b>	chapoutot@ensta.fr

## Internship offer

**Topic of the internship (title)** Automatic test generation for functional software coverage

**Proposed dates of the internship** **Start:** 2019-12-02 **End** 2020-05-29

**Scientific and academic objectives of the internship (detailed description of the internship content, work expected from the intern and expected outcomes):**

1. Context: Software whose malfunctions lead to lose important human or financial resources are qualified as critical. These software are used in all current sectors of activity: in the transport sector, in driver assistance systems or in autonomous driving; in the energy sector, with automatic regulation of energy distribution networks; or in the defence sector. In all these areas, system designers must follow the best practices in software development and implement verification procedures demonstrating that the software does well what it has to do. The software testing activity is an essential component in software engineering and essential in demonstrating the safety of critical software. This activity is based on software specifications to demonstrate, through well-chosen test sets, that the software respects, or at least does not violate, the specifications. The objective is to show that the code does what it is supposed to do and not more. The main difficulty is to measure through the different sets of tests that the specifications are fully covered. In general, measures on constructions of the source code (decisions, branches, etc.), we talk about structural coverage, are implemented to decide whether all functional cases are covered by the tests. The difficulty of measuring functional coverage is due to the difficulty of origin in the way specifications are written, which is generally in natural language. This choice leads to ambiguous specifications. These ambiguities result in implementation choices that do not correspond to the functionalities that should be coded.

2. Work to do: The aim of the internship is to define a methodology for covering functional software tests by automatically generating test sets. To do this, it will be necessary to define a methodology for writing specifications using structured natural language or temporal logic that allows the mathematization of specifications. A state-of-the-art study will be conducted to choose the best formalism for describing the specifications. Then, an algorithm for automatically generating test sets will be defined (which may be based on SAT/SMT solvers for example) and functional coverage will be based on structural coverage metrics.

**Name of industrial partner** Naval Group

**Role of the industrial partner in the internship project** Naval Group will provide a representative piece of software as a case study on which developed method for automatic test generation will be applied on.

**Main contact at the French industrial partner** laurent.raffaelli@naval-group.com

**Targeted Australian university** Any

## Expected profile of applicant

**Level of study** Master's student

**Discipline** Computer Science

**Required qualities, knowledge and skills** Programming and algorithmic skills in functional language such as Ocaml or object-oriented language such as Java Knowledge in formal logic or formal verification methods may be helpful.

