**Résumé du projet de recherche (Langue 1)**

This research proposal is at the intersection of several research areas: combinatorics and probabilities, random generation and model checking. Nowadays software systems are of increasing complexity and scale, important factors of this complexity relating to parallelism and concurrency. The domain of model checking aims at ensuring that such complex systems are free of defects by verifying logical properties on the systems. This has been addressed by many researchers, with various analysis methods and techniques for giving formal complete proofs of correctness. The central point of most of these approaches is to tackle the so-called "combinatorial explosion problem", a phenomenon due to the multiplicity of concurrent executions. However, even if great progress has been achieved, the formal verification of even medium sized software systems mainly remains inaccessible. A very promising approach to scale the existing techniques to larger and more realistic software systems is to adopt a probabilistic view of the problem. Statistical checking allows to deal with larger models and more complex specifications. However the answer is probabilistic and maybe false. Thus statistical model checking requires first a probabilistic method to evaluate the number of trials, and second a stochastic model for uniformly generating the objects. Our PhD proposal adopts a combinatorial approach of developing probabilistic analysis techniques for concurrent systems, based on relationships between the theory of concurrency and trace monoids. It aims at developing theoretical studies as well as an algorithmic framework for practical applications.

**Résumé du projet de recherche (Langue 2)**

The random generation of combinatorial structures is a very rich and ancient domain in algorithmic combinatorics. It can be seen as the simulation of discrete objects, and is highly useful for elucidating properties of objects. Random generation can be used for simulation of discrete structures for many purposes: in testing conjectures and estimating the cost of certain quantities of interest; in the visualisation of asymptotic properties; in comparing different models and evaluate how they fit with real-world data; in testing the correctness and assessing the performance of software applications. We shall concentrate on the last two objectives: recent methods, which allow efficient generation of complex objects, extend the scope of random generation to application domains, where automatic large-scaled generation is needed. So the objective will be to describe validation and benchmarking tests of model checking in terms of structural objects. We shall approach the problem of simulating complex discrete models by following two complementary paths: Boltzmann models and Markov chains perfect simulation. Boltzmann models [6] are originally tightly linked to analytic combinatorics [7] via combinatorial decompositions and recursive equations on generating functions. Markov chains simulation [8] (typically the Propp-Wilson technique of coupling from the past) are adapted to certain types of objects amenable to suitable transformations – instead of being decomposition based, and they are usually linked to certain definite dynamics. Comparing these two categories of models in cases where both are applicable is a challenge and certainly opens a fertile area of research.