Acronyme / Acronym	**MARMOTE**
Titre du projet	**Outils et environnements de modélisation Markovienne**
Proposal title	**Markovian Modeling Tools and Environments**
Axe(s) thématique(s) / theme(s)	☒ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5
Type de recherche / Type of research	☒ Recherche Fondamentale / Basic Research ☐ Recherche Industrielle / Industrial Research ☐ Développement Expérimental : Experimental Development
Coopération internationale (si applicable) / International cooperation (if applicable)	Le projet propose une coopération internationale / International cooperation with: ☐ avec un ou des pays spécifiquement mentionnés dans l'appel à projets / countries explicitely cited in the call for proposal ☐ autres pays / other countries

Aide totale demandée / Grant requested	881267 €	**Durée du projet / Project duration**	48 months

1. EXECUTIVE SUMMARY

The aim of the project is to realize a modeling environment dedicated to Markov models. One part will develop the Perfect Simulation techniques, which allow to sample from the stationary distribution of the process. A second one will develop parallelization techniques for Monte Carlo simulation. A third one will develop numerical computation techniques for a wide class of Markov models. All these developments will be integrated into a programming environment allowing the specification of models and their solution strategy. Several applications will be studied in various scientific disciplines: physics, biology, economics, network engineering.

2. CONTEXT, POSITION AND OBJECTIVES OF THE PROPOSAL

2.1. CONTEXT, SOCIAL AND ECONOMIC ISSUES

Modeling dynamical systems is a basic activity in science and engineering. Among the wide variety of formalisms available for this, this project will focus on "*Markov models*". We encompass in this terminology families of stochastic processes with the Markov property, which have a discrete component: either the space is discrete, or changes of behavior occur according to some discrete-event process.

Markov modeling is used in many fields of science. To name just a few: statistical mechanics [80, 81], evolutionary biology [33], population dynamics [56]. Stochastic modeling is gaining influence in the biological and health sciences. The presence of Markov modeling in Engineering is also widespread, due to its applications in the safety of systems (transport, energy, health), climate change, finance, computer & network engineering, artificial intelligence, operations research, planning and general decision support.

"Solving" a Markov chain model consists generally in applying one of two techniques. Given a quantity of interest in the system that is modeled (a "metric", usually the distribution of some random variable, averages, variances etc.), one may: 1) perform calculations using formulas derived from the theory: we will refer to this as *numerical simulation*; 2) perform computer simulations of trajectories of the stochastic process, followed by statistical estimations: this is commonly referred to as a *Monte Carlo simulation*. Numerical simulation provides very accurate results, but is usually limited by the algorithmic complexity of formulas, and is difficult to apply to very large state spaces. Monte Carlo simulation can easily circumvent this limitation, but provides results of statistical nature, less precise.

We identify two blocking points that limit a practical and routine use of Markovian modeling techniques by scientists and engineers:

- the simulation of "complex systems" of large size, may be too slow and/or provide insufficient guarantees; this is particularly crucial when simulating the *stationary* distribution of the system;
- the existing solution tools are not easily accessible.

Indeed, the simulation of stochastic systems raises several difficult problems when seeking accurate about systems performance for reliability or for validation. There, we face two problems: the size of the space (extremely large) and the numerical values (very low). To quote just one example from the domain of industrial safety, some systems are required to have a 5'9 availability or higher, which means that the system is operational more than 99.999% of the time. It is rather difficult to achieve such measures by the simulation of stochastic models with some guarantee or even confidence in the results.

2.2. POSITION OF THE PROJECT

Position with respect to the context and issues. The ambition of this project is to address the two limitations identified above with two main theoretical and technical developments:

- improve Monte Carlo simulation techniques, in speed and accuracy;
- provide a modeling environment making simulation techniques (Monte Carlo and Numerical) accessible to the general Markov modeler.

These developments will be validated through the analysis of "applicative" models from different scientific disciplines.

The improvement of simulation speed and accuracy will be obtained by addressing the following questions:

- how to sample *exactly* from the stationary distribution of a Markov model, in controllable time? How to reduce or eliminate the dependency on the initial state in the simulation?
- how to efficiently use parallelism in the construction of random samples of the Markov model?

These questions pertain to the fields of Parallel Algorithmics as well as Applied Probabilities, and are theoretical and practical bottlenecks. We discuss below that the partners of this project possess the proper tools and experience to tackle these problems. The driving idea is to exploit *qualitative* properties of the model (monotony, internal structure), to provide better solutions to *both* questions above.

Positioning with respect to other projects Some partners (INRIA/Mescal, UVSQ/PRiSM, TSP/Samovar, UPEC/LACL) have worked together in ANR-06-SETI-002, CheckBound project Model Checking for the Performability and Safety of Computer Systems. Model Checking by simulation has been initially proposed for time-bounded path properties. In CheckBound, it has been shown that the steady-state measures for very large scale monotone systems can be checked by perfect sampling. In the present project, we will be interested in verification of time-unbounded path properties, and non-monotone systems.

Several partners have been recently involved in projects related with Markov Decision modeling for dynamic optimization. INRIA/Maestro was involved in the ANR Multimedia VOODDO project, for optimization of video download and navigation. INRIA/Maestro, INRIA/Trec and UPMC/LIP6 are involved in the INRIA Action Recherche Collaborative (ARC) OCOQS that studies the structural properties of controlled systems. The MARMOTE proposal is complementary since OCOQS is rather focused on the qualitative properties of the control than quantitative figures of Markov chains that are sought here.

Positioning within the CFP. The present proposal fits the CFP in the following dimensions:

- With respect to the objectives of the programme (Section 1.2, pp. 6–7): *Determine today's Software Environments*, *Reinforce our position on the whole chain of software*;

  The ambition of this project is to set the framework for a unique modeling environment for stochastic modeling, populated with advanced solution techniques.

- With respect to the objectives of the call 2012 (Section 1.3, p. 7): *strongly foster projects with a pluridisciplinary character*; concerning "platforms", *structure communities*.

  The project brings together several active groups of the field, total ling in an unprecedented expertise for a project of this nature; the software development effort will require that these

groups collaborate more closely; other groups, in France or abroad, should get interested and eventually join the development effort. The project explicitly aims at providing tools usable by all scientists; it includes several application tasks in several disciplines, and will seek more multidisciplinary contacts.

- Thematic Axis 1: Modeling and Simulation of Complex Systems, with keywords: modeling, simulation, handling of uncertainties, aid to decision and control. Secondary Axis 2: Design and Optimization.

  The proposal is right at the heart of this topic. It will address a family of models, namely, Markov chain and discrete-event stochastic models (hence handling uncertainties), which have a large range of applications. The specialty of the consortium (Performance modeling, stochastic Operations Research) is directly linked to decision making. Some of the stochastic models (Markov decision processes, Markov games) are designed for control and optimization. The improvement of *simulation* techniques that will be developed within MARMOTE is aimed at handling large scale, "complex" systems.

International Positioning. Although there exist isolated software packages devoted to *applications* of Markov modeling, there does not exist projects focusing on Markov chains themselves, with the objective to propose ready-to-use programming tools.

2.3. STATE OF THE ART

We give here a presentation of the state of the art for Monte-Carlo and Numerical simulation techniques, corresponding to work packages WP1, WP2, WP3 and WP4 of Section 3.3. The state of the art for the "application" tasks of WP5 will be given in their respective descriptions.

State of the art on Perfect Simulation

Simulation approaches can be efficient ways to estimate the stationary behavior of Markov chains by providing independent samples distributed according to their stationary distribution, even when it is impossible to compute this distribution numerically.

The classical Markov Chain Monte Carlo simulation techniques suffer from two main problems:
• The convergence to the stationary distribution can be very slow, and it is in general difficult to estimate;
• Even if one has an effective convergence criterion, the sample obtained after any finite number of iterations is biased.

To overcome these issues, Propp and Wilson [88] have introduced a Perfect Sampling Algorithm (PSA) that has been later developed in various contexts, including statistical physics [69, 88], stochastic geometry [74], theoretical computer science [12], and communications networks [7, 37] (for more information, see also the annotated bibliography by Wilson [92]).

Perfect Sampling uses coupling arguments to give an unbiased sample from the stationary distribution of an ergodic Markov chain on a finite state space $\mathcal{X}$. Assume the chain is given by an update function Φ and an i.i.d. sequence of innovations $(U_n)_{n\in\mathbb{Z}}$, so that

$$X_{n+1} = \Phi(X_n, U_{n+1}). \tag{1}$$

The algorithm is based on a backward coupling scheme: it computes the trajectories from all $x \in \mathcal{X}$ at some time in the past $t = -T$ until time $t = 0$, using the same innovations. If the final state is the same for all trajectories (*i.e.,* $|\{\Phi(x, U_{-T+1}, \ldots, U_0) : x \in \mathcal{X}\}| = 1$, where $\Phi(x, U_{-T+1}, \ldots, U_0) := \Phi(\Phi(x, U_{-T+1}), U_{-T+2}, \ldots, U_0)$ is defined by induction on T), then we say that the chain has globally coupled and the final state has the stationary distribution of the Markov chain. Otherwise, the simulations are started further in the past.

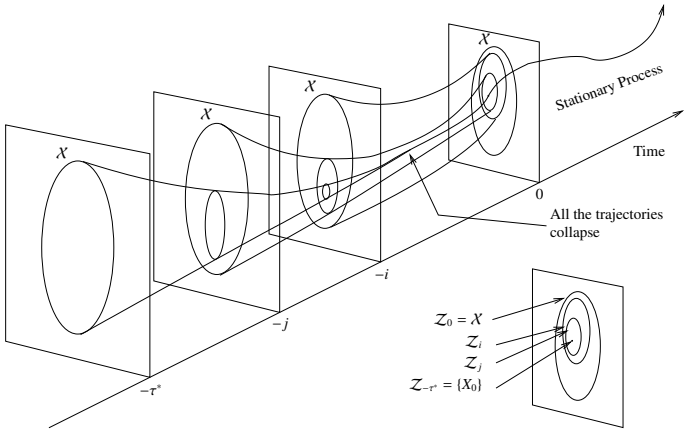


Figure 1: General backward scheme

Any ergodic Markov chain on a finite state space has a representation of type (1) that couples in finite time with probability 1, so Propp and Wilson's PSA gives a "perfect" algorithm in the sense that it provides a *unbiased* sample in *finite time*. Furthermore, the stopping criterion is given by the coupling from the past scheme and knowing the explicit bounds on the coupling time is not needed for the validity of the algorithm.

However, PSA suffers from two drawbacks that jeopardize its applicability:

- The first one is the fact that the coupling time can be very large.
- The second factor in the complexity of PSA is the fact that one needs to run one simulation per state in $\mathcal{X}$, which limits its application only to chains with a state space of very small cardinality.

Coupling time. Some recent work focused on the estimation of the coupling time for certain classes of Markov chains. For example, it was shown in [3, 23] that Markov chains, modeling a class of networks of queues with finite capacities, have a quadratic coupling time with respect to capacities.

In [41], we proposed a new method to speed up perfect sampling of Markov chains by skipping passive events during the simulation. We showed that this can be done without altering the distribution of the samples. This technique is particularly efficient for the simulation of Markov chains with different time scales such as queueing networks where certain servers are much faster than others. In such cases, the coupling time of the Markov chain can be arbitrarily large while the runtime of the skipping algorithm remains bounded. This was further illustrated by several experiments that also show the role played by the entropy of the system in the performance of the skipping algorithm.

Reducing the number of trajectories. Various techniques have been developed to reduce the number of trajectories that need to be considered in the coupling from the past procedure. A first

crucial observation already appears in [88]: for a monotone Markov chain, one has to consider only extremal initial conditions. For anti-monotone systems, an analogous technique that also considers only extremal initial conditions has been developed by Kendall [74] and Häggström and Nelander [69].

To cope with non-monotonicity, Kendall and Møller introduced in [75] the general idea of two bounding processes sandwiching all the normal trajectories, reducing the number of processes having to be computed to two. They applied the sandwiching technique to perfect sampling of point processes, together with the idea of a dominating chain that allowed to handle the infinite state space. Huber [72] also introduced a similar idea of bounding chains for determining when coupling has occurred. In his model, the bounding chains are evolving in some bigger state space than the target Markov chain. However, both Kendall-Møller's and Huber's constructions are model-dependent and do not have an algorithmic counterpart so that they are not straightforward to apply in general.

In [16], we have proposed an algorithm to construct bounding chains (called envelopes), for the case of a Markov chain on a lattice. In [15], we showed that this new approach is particularly effective when the state space can be partitioned into pieces where envelopes can be easily computed. This is the case of many Markovian queueing networks. The monotone PSA and the envelope technique have been implemented in a software tool PSI2 [14, 45].

In some applications, we only need bounds for performance indices. In that case, it is possible to construct a monotone bound for the original chain, and then use the monotone PSA algorithm to sample the steady-state distribution of the bounding chain [32].

The envelope and monotone bound approaches need the lattice structure of the state space. Some non-monotone systems however can be proved to have some extremal states that are sufficient to perform perfect sampling. For example, extremal markings can be exhibited for a large class of Markovian event graphs. In [11], it is proved that when the extremal markings couple, then the whole system couples.

Saving time with parallelism

Simulation of stochastic models is inherently a sequential problem as it seeks to build a path described by: $X(t+\delta) = f(X(t), U(t))$, where $X(t)$ is the state vector at time t, f any function describing the model and $U(t)$ a sequence of random variables whose distributions depend on the model. This is typically a finite order Markov model. The components of vector X may describe a limited influence of the past of the process.

The δ value may be equal to a constant for a discrete-time simulation guided by a clock, a small time interval for a continuous-time approach generally associated with PDE, or the realization of a random variable (in the case of a discrete event approach). Note that we will only focus here on approaches based on discrete-time and discrete event. The goal is to get a very large number of samples in order to prove under reasonable statistical assumptions some guarantee on the performance or the reliability or more generally on a property checked versus some probability thresholds. The simulation consists in

- Initialize $X(0)$;
- Iterate the fundamental equation until it reaches a stopping criterion (timestamp or convergence of an estimator, or coupling, or regeneration or confidence interval...);
- Compute rewards from the resulting path to obtain estimates.

The algorithm is inherently sequential and it is really difficult to parallelize the computations [65].

The most common approach till now is based on a spatial decomposition approach and obey the same principle as distributed algorithms. Indeed the distributed simulation has to compute in parallel the same sample-path that one obtains in a sequential manner. This is typically the way distributed algorithms works but this is not really efficient for simulation.

Suppose that the vector has N components and that we have N processors as well. The idea consists in assigning to each processor one component of the vector for the calculations in parallel. Furthermore we can exploit a principle of locality: Component i depends at time t of the past of the same component and only a few other components. It is not always needed to propagate vector X and make a synchronous computation. Furthermore, we consider a discrete event simulation and not a discrete-time simulation. This practically means that system components do not change at every time clock. It is therefore sufficient to only keep for each component the only moments in which they change (discrete events).

In the context of discrete event systems, this implies that one has only to propagate changes of the various components of vector X. The contribution of distributed algorithms is to verify that asynchronous messages arrive on time (i.e. before they are needed, so in the future of the local clock simulation). Basically there are two approaches: a conservative one, which forbids time faults (messages which arrive too late, i.e. after the local clock) and an optimistic one, which repairs the sample path using rollbacks. These approaches were the subject of numerous studies but they remain of little usage in industry because of lack of guarantees on its effectiveness on parallel machines because of rollbacks. Note however that the conservative approach is implemented in the HLA (High-Level Architecture) RTI (RunTime Infrastructure), a quasi-standard used for distributed simulation for training.

Moreover, the approach by constructing trajectories and calculating an estimator on a finite trajectory does not solve the problem of measuring stationary. It is necessary to use strong stochastic result (for instance regeneration) and typically only heuristics are used. Perfect simulation provides a much better way to analyze the models.

A radically different approach consist in dividing the time horizon $(0, T)$ into intervals (t_i, t_{i+1}) that will be simulated in parallel (with $t_1 = 0$ and $t_{N+1} = T$). The simulations are performed independently and in parallel. This is the time parallel approach. There is however an obvious problem of consistency: the time slot number (i) must start from the final state of the time slot number $(i - 1)$ that by definition it is not known initially. We speak of spatial fault when the two parts of trajectories do not meet.

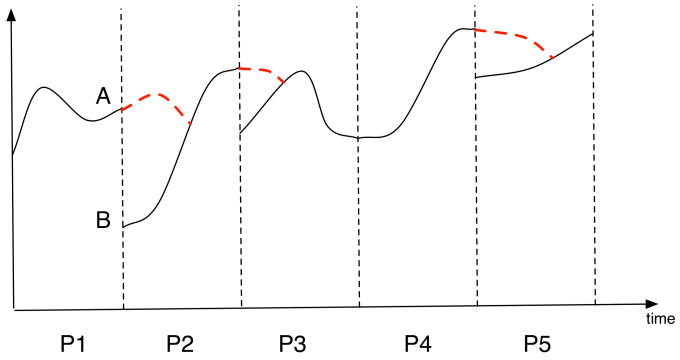
At first, suppose we glue together the fragments obtained in parallel but not yet guarantee the consistency of the trajectory in its entirety. There are several possible approaches based on qualitative properties of the model:

- Partial predictability: assume that a simple inspection of the sequence of inputs reveals some points of the sample path without running the simulator. These points will then be starting points for correct paths calculated in parallel [64].
- The dynamical model is associated with an induction based on an operator in an associative algebraic structure (i.e. a semi-ring). It is then possible to use Algorithm “Parallel-Prefix” to obtain a correct efficient implementation [8].
- Coupling: we make a random choice for initial states and correct the spatial errors in a later

phase of iterative corrections and recovery simulations [35, 50].

If we are able to find some of the points $(n, S(n))$ by inspecting the trace input I, we say that the model is partially predictable and parallel simulations may be started from these remarkable points. This approach is then very effective as long as one is able to find many points and they are regularly spaced. Somewhat surprisingly, many models of traffic in a network partially prove predictive. Typically, some congested states or starvation can be predicted by a simple inspection of the traces of arrivals.

When the model is used to write the state vector at time $(t+1)$ as the application of an associative operator on the state vector at time t, then the calculation can be done in parallel and without spatial errors using T processors for $log_2(T)$ steps using the method "Parallel-Prefix" to simulate the trajectory of T dates. This allowed for example to parallel simulation time for some classes of Petri nets or queuing.



Finally, when both approaches fail, it is still possible to correct spatial errors. We start by fixing the cardinality of the number of parallel simulations. Initial points are taken at random for each simulation except the first one, which is correctly initialized. Time intervals are executed in parallel and then are corrected, if necessary. During the correction phase, we use the same random sequence to control the trajectory. This means that we will have the same length of service and the same process happened if we simulate a queue for example. This will allow to link the trajectories and to the speculative calculation.

Since we replay the sequence of inputs, when two paths merge, we have a coupling of the two simulations and it becomes unnecessary to continue the second simulation because the two paths merge for the rest of the simulation. This is exactly what is defined in the literature as a stochastic coupling. The coupling is used to validate the simulations already carried out in an earlier phase for a time segment in the future. For example, in the figure above, there is coupling in phase 2 of all parts of the trajectory. The trajectories of the first phase are shown in black solid lines and those of the second dashed red. In this example we built an exact path in two phases with 5 processors.

In the best cases, one phase of correction but the number of corrections phases can be as long as the sequential approach in the worst case, because $N-1$ steps may be necessary, if we do not couple and that you must always repeat the parallel simulations.

Finally, there are strong limitations in the time parallel approach as well:

- To partially predictive models, one can prove the existence of these points but nothing is known on the frequency of such points. Therefore, before the examination of the input trace,

we do not know about the number of interval timing and their respective length. Indeed it is necessary to have a high degree of parallelism to get a large number of time intervals with approximately constant size to prevent a processor lag, which penalizes the overall calculation.

- The approach related to "Parallel-Prefix" implies that one is capable to prove such an evolution equation and such a result is relatively rare.
- For the approach based on the correction, if we do not couple enough, we finally get the same simulation time than the sequential approach and there is no gain.

Numerical simulation of Markov Chain models

The modeling power of Markov models can be attributed to the fact that many computations (probabilities, moments) can be expressed as linear algebra operations. The typical situation is to compute the stationary distribution of a Markov chain. This consists in solving a fixed-point linear system of equations $\pi = \pi\mathbf{P}$. Numerous researchers have studied the ways to compute or approximate efficiently the solution to this equation, yet new techniques are still proposed. In particular, we have proposed in [13] new iterative algorithms for computing approximations with a known error.

Computing non-stationary (transient) distributions for Markov models has relatively received less attention, but is receiving more and more attention, due to the numerous application possibilities (*performability* in Engineering, finance, evolutionary biology, etc.) We have proposed in [10] a model of *Semi-Markov Additive Processes* (SMAP) which turns out to generalize many specific models previously proposed in the literature. Its definition starts from the time instants of a semi-Markov process, $\{T_n; n \in \mathbb{N}\}$, based on an underlying ("environment") Markov chain $\{Z(n); n \in \mathbb{N}\}$. It assumes that quantity $Q(t)$ accumulates 1) at jump times of the underlying Markov chain, and 2) according to a Levy process between jump times. It turns out that the joint distribution of the quantity $Q(t)$ and the state of the underlying environment, can be captured by its space-time Laplace transform $\mathbf{K}$, in simple linear fixed point equations of the form: $\mathbf{K} = \mathbf{A}+\mathbf{BK}$, where matrices $\mathbf{A}$ and $\mathbf{B}$ are built from the parameters of the evolution and the accumulation processes. In addition, the distribution of *level crossing times* satisfy similar equations in certain cases. This simple structure allows, at least in principle, a *uniform treatment* of a huge variety of situations, including Poisson and Wiener processes, Markov and Batch-Markov arrival processes [85], Markov-Modulated Poisson Process (MMPP) and Rate Process (MMRP), and the simple Markov and Semi-Markov chains.

We have applied this model to specific situations, and for specific performance metrics: in [10] in the context of video distribution, and [24] in the context of Evolutionary Biology: this illustrates the potential for multidisciplinary applications of this model. Exploring systematically the potential of the model remains to be done.

The SMAP model is not the first to be proposed (various Markov or Semi-Markov Arrival or Reward processes have been used for a long time), and not the most general (Markov Regenerative processes of Cinlar [57, 76] are more general) but we believe it realizes the adequate challenge between power of expression and numerical complexity. WP 4 is devoted to developing the numerical techniques for this. The techniques that will be investigated are: formal and numerical inversion of Laplace transforms (in particular, the well-known algorithm of Abate & Whitt [48]), analytic and Laurent expansions of matrix functions (*e.g.* Avrachenkov & Lasserre [51]) for

moments, tensor methods for correlations (Nemirovsky [84]), and complex-variable asymptotics (Bender [53], Flajolet & Sedgewick [63]) for large-time or large-space approximations.

Modeling environments

The idea of providing users with software environments to help them realize their modeling experiments is, of course, not new. The most popular mathematical modeling environments, such as Matlab, Scilab, Mathematica, Maple, all provide packages with functions specialized to families of models (e.g. Matlab's SimuLink for system modeling), or analysis techniques (e.g. statistical packages). There does not exist standard packages specifically devoted to modeling with Markov chains. In the specific context of Queueing Theory (which is largely based on Markov Chain modeling and is familiar to the participants to this project), the list of software maintained at [70] is typical of the fact that there is a large variety of solutions techniques adapted to specific models or families of models, but also that there is no uniform presentation or unique access point to these tools, that would help the newcomer to apply one to his/her needs. On the other hand, analysis tools are sometimes presented as "packaged" behind a graphical user interface (e.g. [58]); in that case, these tools cannot be used outside of their application context. But there is a need for libraries giving access to efficient numerical algorithms at a lower programming level.

The participants to this project have a long-term experience of software realizations and modeling environments. The INRIA/Maestro partner participated to the ESPRIT II project IMSE (Integrated Modeling Support Environment, 1987-1991) which was devoted to the integration of simulation and solution techniques for Queueing Networks, and led to the tool Modline, marketed by the (late) Simulog company. Partners INRIA/Maestro and INRIA/Mescal have worked on the ERS environment for Discrete Event Systems and (max,+) modeling [38]. The UVSQ/PRiSM and INRIA/Mescal partners have been developing the Xborne [25], PEPS [9, 42] and Psi [45] environments for, respectively, the construction of bounds, the numerical solution and the simulation of Markov models.

2.4. OBJECTIVES, ORIGINALITY AND NOVELTY OF THE PROJECT

In the course of the project, we shall set up the structure of a *software environment* able to give access to the most efficient solution algorithms for Markov models. This environment will not be complete at the end of the project; but will be conceived as *open* as possible so that capabilities that are missing will be later filled by a community of contributors. The environment will nevertheless be populated with a wealth of standard algorithms, which will serve later as benchmarks, with *new numerical methods* developed in this project, and with enhanced *simulation algorithms* developed within the project. The class of Markov models addressed will be as large as possible, with an emphasis on *formal* specifications which will allow for an automated discovery of the qualitative properties exploited in clever solution algorithms (including simulation, see above).

Finally, we will realize the "proof of concept" by exploiting the scientific links of this project's partners with scientists of several disciplines, to test theoretical advances and software on concrete modeling situations: in Network Engineering, Biology, Physics and Economics.

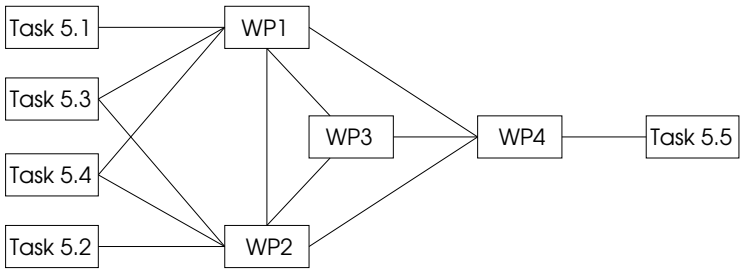
3. SCIENTIFIC AND TECHNICAL PROGRAM, PROJECT ORGANIZATION

3.1. SCIENTIFIC PROGRAM, PROJECT STRUCTURE

The project will be organized with 5 main work packages, and a management work package WP0:

1. WP 1: Perfect Simulation for non-monotone systems
2. WP 2: Parallel Simulation: the monotony approach and its benefits
3. WP 3: Development of a modeling environment for Markovian systems
4. WP 4: Numerical Simulation methods for Markov models
5. Application work package WP5, divided into 5 tasks, namely, Task 5.1: Network dimensioning for cloud computing environment, Task 5.2: Parallel simulation of the cell and apoptosis, Task 5.3: Stochastic verification by simulation, Task 5.4: Applications to laser physics and statistical mechanics, Task 5.5: Markovian games and economic problems.

The functional relations between these work packages can be summarized in the diagram below. WP1, WP2 and WP4 are in strong methodological interaction, and their result will be implemented in the software WP3. Application tasks will use results of one or two WPs.



3.2. PROJECT MANAGEMENT

The management activities are identified in WP 0, lead by A. Jean-Marie and J.-M. Fourneau, with the participation of each partner's head, thus constituting a "management board". The partners will be linked through a Consortium Agreement. The animation and unity of the project will be ensured by two sets of actions. On the one hand, there will be regular *general assemblies* where advances of methodological, software and application nature will be presented. It is foreseen that two such assemblies will take place per year. Each work package (or task in WP5) will be led by one or two co-responsibles reporting to the management board.

On the other hand, the consistency of software activities (distributed among partners) will be enforced through regular meetings (visioconference) between engineers and researchers of a same task. The organization of "coding sprints" and tutorial events will be studied. Software development will be hosted by INRIA's "forge", thus providing a common development infrastructure. The project will be able to rely on the expertise of INRIA's Software development teams, which can provide project management and technical assistance.

The web site of the project will, of course, gather references to the publications of the project, but will also be a common repository of software and give access to the tools developed (see also Section 4).

3.3. DESCRIPTION BY TASK

3.3.1. WP1: Perfect simulation for non-monotone systems

When the function ϕ used in the fundamental equation ($x_{n+1} = \phi(x_n, e_{n+1})$) used to simulate a Markov chain is not monotone, then the only way one can simulate a Markov chain from the past is to generate one trajectory per state of the Markov chain and run the simulator up to coalescence of all the trajectories. The cost of this general approach is too high to be used for very large chains.

In the past we have developed several approaches to circumvent this difficulty for lattice state spaces. Instead of the original chain, we used two bounding processes (envelopes) and we show that, whenever they couple, one obtains a sample under the stationary distribution of the original chain. We have shown that this new approach is particularly effective in several cases, all pertaining to the networks of queue framework. More precisely, this technique amounts to replace the fundamental equation by a couple of equations:

$$\begin{aligned} M(t+1) &= \sup_{m(t)\le x\le M(t)} \phi(x, e_{t+1}), \\ m(t+1) &= \inf_{m(t)\le x\le M(t)} \phi(x, e_{t+1}). \end{aligned}$$

Starting from the extreme states in the state space, the couple $(m(t), M(t))$ always provide lower and upper bounds on the state $X(t)$. Therefore, whenever m and M meet, all trajectories have coalesced and the simulation stops with a stationary distribution.

This approach has two drawbacks. First, it requires the computation of the supremum and the infimum of ϕ over all states in the interval $[m, M]$. In general the complexity of these computations is linear in the size of the interval $[m, M]$, so that the new approach has the same complexity as the exhaustive one.

The second drawback is that the coupling time of the bounds is larger than that of the original chain, and may even be infinite.

Sub-task 1.1: The goal of this sub-task is to propose new algorithms to circumvent these two drawbacks. This will be done both at a structural level by proposing analytical tools (to compute the optimal splitting time, for example) as well as at the application level where ad-hoc approaches will be used. Already today several tracks have been opened. A first possible direction of research is to extend the state space to make supremum and infimum computations easy. A first attempt at this can be found in [34].

Actually, there is little hope that substantial improvements can be obtained in the general case. A more promising perspective is to exploit semantic and geometric properties of the event space and the state space to derive supremum and infimum computations, as done in [15] for the specific case of queueing networks.

The second problem comes from the coupling time (possibly infinite for the envelopes). It can be overcome by generating all trajectories contained in the interval $[m, M]$ as soon as the size of the interval is manageable (this technique is called splitting in [16]). The challenge here is to find a good splitting time ensuring both a fast coupling and a reasonable number of simulated trajectories.

Another direction of research is the construction of a bounding process whose stationary distribution is known and construct an upper (resp. lower) bound for the upper (resp. lower) envelope M

(resp. m) by following a reversed trajectory for the bounding process. This technique is currently being investigated in monotone systems but is promising for non-monotone systems as well.

Sub-task 1.2: Here we list some challenges to go beyond the current setting for perfect sampling. This task has higher risk than Sub-task 1.1.

The structure of the state space. Most of the previous results on perfect sampling assume a lattice structure of the state space. This is a natural assumption for many queueing systems. However, in closed queueing systems or Petri nets, the constraint on constant total number of customers breaks the lattice structure of the state space. We would like to better understand and extend the ideas of extremal markings in Markovian event graphs from [11, 36] that do not rely on some ordering of the states that is preserved by the dynamics of the system.

Coupling time. The second direction is to study the coupling time for some specific models, such as for instance the class of almost space-homogeneous Markov chains defined in [15]. Only some partial results are known for Jackson networks with finite capacities [3, 23]. The approach would combine the recent developments on mixing times for Markov chains [79].

Random graphs. Finally, another task would be to study larger classes of systems than networks of queues or Petri nets. For instance in graph theory, some research has already been pursued in random generation of combinatorial structures (configuration model graphs [59], Exponential random graphs [55]), or independent sets [60, 61]...). The goal would be to apply the new techniques we will develop in Task 1.1 to those problems. Also, we wish to investigate the relations between the coupling time and the algorithmic complexity of problems. This could explain the impossibility of a good (efficient) sampling for some problems.

Sub-task 1.3: The aim of this task is to provide a simulation kernel for the perfect sampling of non-monotone Markovian systems. This kernel will be compatible with the software environment developed in WP 3 and consequently allow the comparison of methods (semi-numerical and simulation). This will be helpful for the validation of methods and software.

Two previous simulation kernels have already been developed: Ψ [44] for the sampling of Markov chains with a given transition matrix and Ψ^2 [45] for the stationary sampling of networks of finite queues. These software have been used to test the efficiency of new methods with ad-hoc code modifications [14].

Based on these fundamental results a new version, including all the recent improvements should be developed. The challenge is to propose a generic encoding of envelopes transition functions which are at the heart of the simulation (each simulation step need one call to the transition function).

Moreover, to reduce the simulation time, the skipping technique introduced in [41] modifies the basic data structure used to store intermediate trajectories. If well encoded this technique will be particularly effective when events occur on several time scales, which are usually difficult to solve. The difficulty is to change dynamically the relative probability of events. The challenge here is to use cost amortizing techniques (for example by constructing a potential function over the simulation cost).

Finally, the code will be parallelized and tested on two strategies :
- the statistical independent replications which leads to an improvement on statistical estimators and then reduce the simulation time. This should be easily implemented and tested;
- the parallelization inside the trajectories as studied in WP2, this will be more complex because

coupling times are related to the macro time that should be adapted.

3.3.2. WP2: Parallel simulation: the monotony approach and its benefits

The limitations come from too much respect of the constraints of distributed computing and forgetting about the randomness of the models we want to simulate. We will therefore develop new methods for simulating stochastic models suitable for parallel machines. These new methods will take into account qualitative properties of the stochastic models and simplify the parallelization of the simulation. Most of these properties are variants of an intuitive concept of monotony (or anti-monotony) that we will detail later. Specifically, we will study the following:

- Event monotony
- Sequence monotony (monotony related to the traces)
- Monotony with respect to input variables.

These methods will be applied first for time-parallel simulation but the optimistic space decomposition approach will also be studied in this WP in cooperation with the application work package WP5. Note that the difficulties are somehow symmetrical. Parallelization of space is made difficult by the problems of temporal errors (the distant events arrive in the past of a local schedule) while the parallelization of time involves dealing with the potential problems of spatial errors (the system state is not consistent at some time stamp).

We have already shown how to combine the approaches of predictability and ex-post corrections proposed by Mitrani, Nicol and Fujimoto [65] with the monotony properties of random models to obtain easier to parallelize simulations (with the time parallel approach). More precisely, we have proved [27, 28] two innovative approaches related to two concepts of monotony for random models that we now present.

First we need to compare sample paths: they are denoted as sequences of inputs and outputs. To simplify the presentation we assume that we use a point-wise comparison. Indeed, this type of ordering on sequences is consistent with positive rewards applied at each timestamp and several rewards used for reliability, performance, model-checking are positive.

The key idea follows: simulation is seen as an operator acting on a set of hidden values (initial states) and an input sequence to produce an output sequence. Note $O = M(I, a)$, where O' is the output sequence, I is the input sequence and a is the initialization of the simulation. Therefore one can define two monotony properties. Both of them are much simpler to prove that the stochastic monotony [29].

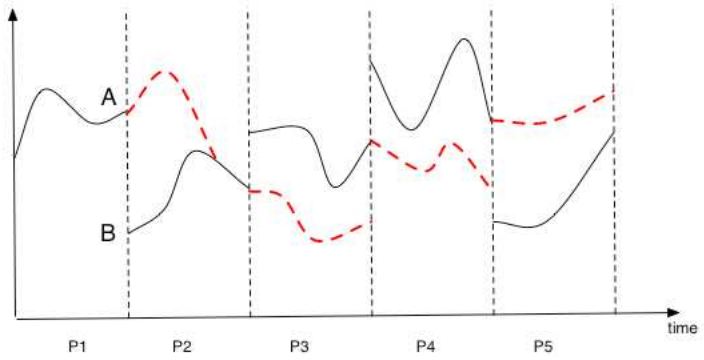
Let M be a simulation model, M is input-sequence-monotone ("inseq-monotone" for short) for orderings α and β if and only if, for all input sequences $I1$ and $I2$ comparable with α ordering, then the output sequence produced by the model M for $I1$ from any initial state a is comparable, with the ordering β, to the output sequence produced by model M for $I2$ from the same initial vector a.

Similarly we say that a simulation model M is hidden-variable-monotone ("hv-monotone" for short) for ordering γ and β if and only if, for all vectors or initial states a and b comparable with the ordering γ and any sequence of inputs I, the output of the model starting from a is less than the output of the model from b where the comparison is made with ordering β (both simulations use the same sequence I).

Our first approach involves partially predictive models: if some points of the models can be predicted from the trace, we modify the input sequence by proving that it causes an increased frequency of predicted points, and that the sample-path obtained is a bound of the exact trajectory. This results in a more easily computable bound because it is easier to parallelize. This bound is often sufficient, since we try to prove a guarantee on the performance or the reliability of a system. This is also typically what we need for stochastic model checking.

Consider the following case. Assume that one uses a natural order on the states. Suppose we can predict state 0. We compute a trajectory on a lower bounding input sequence. The inseq-monotony of the sequences implies that the sample path is a lower bound of the exact trajectory. Changing the input sequence makes it more predictable because it may more often go through state 0. Another approach is to prove that an error on the initial state of a time interval impacts the output by providing a bound of the whole path [26]. It is no longer necessary to recalculate the true value if the bound is sufficient to prove the guarantee. This decreases the time required to build a complete path. In the usual approach, the coupling allows us to avoid to compute again the trajectories and it ensures that part already computed parts are correct. In our approach, we always use the coupling but we add the comparisons of paths. Suppose we want to get an upper bound of the trajectory (the analysis can also be made for lower bounds as well but fixing the type of bound help to describe the details).

At the end of a calculation step, we check the correction of the parts of the trajectories. If the path of the segment i was started in a state larger than the final state of the segment $i-1$ and if part $i-1$ is correct, then the hv-monotony proves that the trajectory calculated on part I is an upper bound on the exact sample-path. It is therefore correct and it is unnecessary to compute it again. For instance, in the following figure, we obtain a proved upper bound of the sample path after two rounds if the model is hv-monotone.



Furthermore, one can make some speculative calculations, which are not possible in the classical approach. Assume that we want to compute a lower bound of the path.

Assume that at step k, the two parts of trajectories i and $i+1$ are joined in U, but they are not yet consistent because part i is not consistent. Parts i and $i+1$ must be recalculated. But there is no need to recalculate the trajectory starting in U and one must choose another initial point. To provide a lower bound, we take a random point smaller than U (named V). Indeed this speculative computation increases the probability to end the correction step at the next step [31]. Note that by taking V equal to the minimum of the state-space (if it exists), we are sure that the algorithm for a

lower bounding sample-path converges within one iteration. Therefore we can derive algorithms with a proved convergence within a fixed number of steps of correction. This provides a trade-off between the accuracy and the time needed to obtain a bound. Again, this is not possible with the usual approaches.

- Clearly we use qualitative properties of the model to avoid synchronizing the simulations time segments. This provides a much higher efficiency for the processors. We have published some parts of the theory and some applications since 2009 [22, 30]. But there are still numerous issues to resolve both theoretical and practical. And we need to develop an efficient implementation on a GPU. Indeed in the time parallel approach all the processes execute the simulation model at the same time on distinct parts of the input sequence. This is typically the type of program convenient for GPU. We specifically want to address the following topics:

- Study the influence of dimensionality. In a high dimensional space with a partial order, it is unlikely that any two points are comparable. The question is to check to what extent this limits the usefulness of the computation of bounds. One can choose a bounding point which dominates much of the space-space and several heuristics are possible.

- Prove an adaptive algorithm that changes the random initializations to achieve a convergence in K iterations through strategies based on the past and the time- segments already consistent.

- Combine the bounding approach with the "Parallel-Prefix" technique. Is it possible to bound a complex simulation model by another one, which allows a parallel prefix computation?

- Develop an equivalent theory for distributed simulation based on optimistic space decomposition (the approach proposed by Jefferson). Is it possible to derive a bounding trajectory, which does not require to fix the time faults? Which type of qualitative properties do we need? We believe that monotony can bring a solution to this problem.

- Derive some algorithms that compute both upper and lower bounds. Indeed if a part already computed is not consistent for an upper bound, most likely it is consistent for a lower bound. Therefore, computing both bounds at the same time allows to take advantage of most of the computed time segments.

- Study a memory allocation mechanism to store the parts of the simulation before they are glued together. Indeed, as many speculative computations are possible before we converge, one has to store them and find an efficient algorithm to check the consistency of all the segments.

- Derive an iterative method to dynamically improve the trace to increase the number of predicted states. First, if the modified trace does not change the number of predicted states in a time interval, one can continue with the initial trace to obtain a more accurate result (the input trace does not change). Second, one can iterate of the trace transformation until a sufficiently large number of points is obtained.

3.3.3. WP3: Development of a modeling environment for Markovian systems

The purpose of this work package is to provide to the general scientific user a "modeling environment" which should give them access to algorithms devised by specialists. This environment will be conceived as open as possible, so that users typically will be able to use their favorite modeling software (Matlab (R), Mathematica (R), Maple (R), or direct C, C++, Python, Java programming). Symmetrically, contributors of algorithms should have minimal constraints. Within the time- and resource-frame of this project, realizing a fully functional platform is not realistic. Instead, the work will concentrate on setting up a solid architecture based on modern participative software development concepts. It will be populated with the minimum user input interfaces (description languages, configuration files, basic graphical interfaces), with the basic solution algorithms taken "from the book" (*e.g.* Tijms [89], Latouche and Ramaswami [77]), and with the algorithms conceived in other work packages of the present project. It is not the purpose of this project to provide a fully packaged, commercial-grade software. The success of the approach should be validated by the adoption of the software by scientists: from WP 5 applications, and others if possible.

A strong emphasis of this project is on the exploitation of "qualitative" properties of the models in order to facilitate the analysis, be it by Monte Carlo simulation (see WP 1 and WP 2) or by numerical algorithms (WP 4). Such qualitative properties include properties of state-to-state transition rules (monotonicity, convexity, submodularity...), as well as structural properties of the transition graph/matrix (quasi-birth-death, reversibility, symmetries...). It is therefore essential that the environment be capable to identify such properties in a formal description of the model. This work package will be realized by software engineers, starting from specifications provided by the members of the whole project (WP 0) early in the course of the project. We can nevertheless identify at this point the following sub-tasks devoted to the development staff:

1. Set up the project's engineering method; gather functionality needs from potential users (see the "Application" work package WP5) and initial contributors; design software architecture, maintain documentation;

2. Develop/collect ontologies for: classes of models, mathematical objects, metrics, methods, results (data representations);

3. Develop high-level modeling languages for the families of models under consideration: Markov and semi-Markov models, both in discrete and continuous time; Markov-additive processes (see WP 4) and their controlled versions.

4. Develop a classification tool, capable to recognize from the formal specification large families of models (e.g. for Markov models: birth-death, quasi-birth-death, reversible, etc.)

5. Develop software modules for new algorithmic techniques [13] and basic and "well-known" algorithms (from bibliographical sources under the guidance of research staff); adapt/embed software already developed by the participants to the project. In particular, the package Xborne [13, 25] developed by UVSQ/PRiSM will be contributed to the environment.

6. Develop classical algorithms dedicated to the computation of the optimal policy in a MDP (and in a SMDP). These will be implemented based on the usual *value iteration* and *policy iteration* and extensions of them.

7. Develop several classical methods to compute Markov Games Equilibria, taken from the literature on competitive MDPs or using "cobweb" approaches. Most of them are a mixture of dynamic programming and Nash Equilibria computations.

8. Perform porting tasks: adapting algorithms to the different target software environments.

9. Set up the web site of the project, allowing in particular a web access to the tools developed.

The environment OpenAlea [47], devoted to Plant Modeling, will be a source of inspiration since it shares many operational objectives with the environment we envision. This is a component-oriented platform embedding contributed software modules written in C or C++ inside a common Python environment. The modeling language itself is interpreted and allows a dynamic interaction with the user. The environment allows the specification of a workflow for the analysis of models. Actually, OpenAlea does include Markovian modeling modules, and the possibility to interface both environments will be investigated from the beginning of the project.

3.3.4. WP4: Numerical simulation methods for Markov models

This work package groups the efforts of several partners on the development of new methods and algorithms for solving (*i.e.* computing numerically quantities of interest) Markov models. The implementation of the algorithms in the MARMOTE environment will be done in WP 3; WP 4 identifies the theoretical part of this development, insofar it is possible do separate it from implementation.

This work package will include the following developments on different categories of Markov models, brought by different partners. Providing the modeling environment with up-to-date algorithms should be a good incentive to attract potential users.

Markov Chains. Current investigations on the computation of stochastic bounds on stationary and transient distributions of Markov chain will be pursued, in particular in connection with the notion of truncation of large or infinite state spaces.

Semi-Markov Additive Processes. The purpose is to develop the numerical methods for the class of SMAP (Semi-Markov Additive Processes), both for *quantities* reached at some time t, and for *hitting times* at which a certain quantity is attained. Various techniques will be necessary for the different relevant quantities (see the state of the art on p. 10): linear algebraic methods for moments, numerical Laplace transform inversion in multiple dimension for distributions, analytical combinatorics and singularity analysis for asymptotics.

Markov Decision Processes (MDPs) and Markov Games. From a numerical point of view, the bottleneck of the existing approaches are the algorithmic complexity and the curse of dimensionality which can occur even for simple models.

Thus a technical algorithmic challenge will be to apply usual methods to models that are specified formally, and not in extensive matrix form. Indeed, the experience shows (*e.g.* in the ANR VOODDO project) that not storing matrices allows to solve much larger models. This comes of course at a cost of increased computing time.

Another technical challenge is to reduce the set of potentially optimal decisions by automatically detect and exploit *structure* (*e.g.* monotony) in MDPs. These structural properties of the optimal policy allow to reduce the set of candidate optimal policies.

Some of the approaches here can be equally extended for dynamic games. Hence, the connection with Task 5.5 will be secured by this development.

Approximate dynamic programming. Unfortunately, it is not always possible to identify and prove structural properties that would allow significant reduction of the set of candidate optimal policies for a specific MDP. Approximate dynamic programming (ADP) [71, 87] is a very active area of applied research on MDPs that uses various approximation techniques to derive close-to-optimal policies, by simplifying either the search space or the complexity of one iteration (in value/policy iteration algorithms). We expect that our results on bounds for Markov chains can give a valuable contribution to this field. Also, many approximations proposed in the literature include simulation of Markov chains. This represents a clear connection with WP 1 and 2, that we expect to yield improved algorithms for ADP.

3.3.5. WP5, Task 5.1: Network Dimensioning for Cloud Computing Environments

Context. Cloud Computing is a recent advancement wherein IT infrastructure and applications are provided as "services" to end-users under a usage-based payment model [90]. Cloud services are broadly classified in three classes named SaaS, PaaS and IaaS available to cloud consumers (residential, enterprise, or organization users) [46]. The operators of these infrastructures benefit from economies of scale and multiplexing gains enabled by sharing of resources of Telecom and Datacenter infrastructures through virtualization of the underlying physical equipments [91]. Cloud service delivery is built from the convergence and the combination of different types of services processed by network infrastructures and datacenter infrastructures. The former deliver connectivity services those are essential to allow accessing any computing and storage services delivered by the later . Different patterns of virtualization are used to abstract the physical resource elements hosted in the network (e.g. IP routers, Ethernet switches, Optical Cross-connect) and datacenter (e.g. computing clusters, storage racks) infrastructures.

Performance evaluation of cloud computing environment for different application models under transient conditions is extremely hard because: (1) clouds exhibit varying demands, system sizes, and resources (hardware, software, network), (2) users have dynamic and competing QoS requirements, and (3) applications have varying performance and dynamic application scaling requirements. In [68], they develop the simulation framework CloudSim that allows modeling, simulation and experimentation of emerging cloud computing infrastructures and application services. A simulation environment has been described in [82], which is capable of processing diversified request dynamically and responding the requests in time.

Objectives of the task. In this project, we propose to study and apply new and efficient simulation methods: perfect and parallel simulation in order to provide dimensioning solutions for massive scale cloud computing system. Both theoretical concepts of simulations and development of software tools in order to test automatically cloud environments will be the goals of this task. To our knowledge, there are no other studies which apply these new simulation methods for cloud computing environment. We need to adapt these simulation methods to the cloud environment constraints, which has specific properties as: high number of events, many kinds of resources and services, dynamicity of the behavior.

If we suppose that the system is stable, then it is clear that as the system is very complex, the computation of the stationary probability distribution is very hard. Due to the large number of events, then simulations require a convergence time which is high and uncertain to estimate the stationary probability distribution. It is clear in this context that the problem arises on the accuracy

of results and dependence on the initial state by conventional simulation. We believe that on this particular aspect, the perfect simulation could provide interesting solutions in terms of accuracy of the results and the achievement of these results in a finite time. Moreover, with the complexity of the applications in terms of resource needs, but also the volume of applications, it results that the amount of application instances will increase. In this context, parallel simulations can improve simulation time and accuracy of performance results. The simulation models will be provided by Alcatel-Lucent and they represent a real architecture we want to study. The networking and computing components will consist in detailed examples of elements of real cloud computing infrastructures provided by Alcatel-Lucent to its customers. We expect to simulate massive scale cloud computing infrastructures in order to show the efficiency of the methods in terms of accuracy and simulation time. We have the following main steps in this task: -Specification of accurate models for applications and networks as given in [68]. The goal is to define models for cloud environments given by Alcatel Lucent. The development of a software tool implementing these functionalities is one objective of this step. -Study and adaptation of parallel and perfect simulation to cloud environment constraints. The transient structure of the system involving high variability of the number of events with the time is an important feature to study with these simulation methods. The question is how these methods could derive accurate results with a better simulation time than other simulation methods (as Monte Carlo method). We need in this step to develop simulation algorithms which take into account cloud computing constraints. -Delivering dimensioning solutions for networks as trade-off between QoS guarantee and cost control. The software tool developed will give automatically the solutions, by taking as input different kinds of network architectures and service applications provided by Alcatel Lucent. One objective is to see the impact of fluctuating resource and service demand on the QoS requirements of the application. It is also important to study efficiency of simulation methods under these conditions related to the dynamicity of the system.

3.3.6. WP5, Task 5.2: Parallel Simulation of the Cell and Apoptosis

Biological context. Several human diseases highlight the dysfunction of the cell linked to a disturbance of the death cycle of the cell. In these conditions, there is a massive cell death by apoptosis (programmed cell death), occurring during acute trauma (heart attack, etc.). Although this cell death is essential during embryonic development and tissue homeostasis for adults, an abnormal regulation of apoptosis can lead to an accumulation or an uncontrolled loss of cells. A deregulation of apoptosis is also observed in many other human diseases such as cancer, neurodegenerative diseases, autoimmune diseases or stroke. This programmed cell death is a highly regulated complex pathway, involving many intracellular agents providing enhancing and/or inhibitory activity during the reaction. The UVSQ/PRiSM group has started to work on this topic in 2009-2010 with biologist G. Lecellier (formerly at UVSQ and now with Université de la Polynésie Française) as part of a pluridisciplinary BioInformatics project financed by the UVSQ.

Objectives of the simulation. The objectives of this Work Package consist in the simulation of the complex biological process of apoptosis. This simulation is based on the implementation of a large amount of heterogeneous actors that appear in the biological networks specific to apoptosis. The simulation will be both qualitative and quantitative. Our approach is to design, develop, validate and use a software platform that will detect and / or confirm classes of proteins, which may be involved in apoptosis. We also use this software to identify new agents and target areas that can modulate or regulate the apoptotic activity.

This biological process is extremely complex and involves a large amount of molecules but also many heterogeneous molecules [78]. Therefore, we will need a large computing power to run the model [49]. In order to use them efficiently, many techniques and approaches developed for high performance computing will be needed. More specifically we want to use an approach based on domain decomposition. Each processor performs the simulation of a part of the cell. The problem is to find an efficient partition of the objects of the model, which can be easily used on our many cores machines.

The overall principle of the simulator is as follows. It can be actually considered as an evolutionary game simulator. Proteins interact to form complexes according to the possibilities of interaction and structural incompatibilities. If the threshold concentrations of the complexes are met, one triggers the increase or decrease in the concentration of its neighboring proteins in the network of transitions. The objective is, from an initial concentration to estimate the likelihood of achieving a level of concentration of one or more proteins, which provokes apoptosis.

3.3.7. WP5, Task 5.3: Stochastic Verification by Simulation

Context. Model Checking is a technique for automated verification of software, hardware and network systems. It has been introduced to verify functional properties of systems expressed in a formal logic like Computational Tree Logic (CTL). It is done by accepting as input system models and the properties or specifications that the final system is expected to satisfy and by giving outputs Yes if the given model satisfies given specifications and No otherwise. Stochastic model checking is an extension for the formal verification of systems exhibiting stochastic behavior. The system model is usually specified as a state transition system, with probability values attached to transitions, like for example Markov Chains. The stochastic model checking consists in deciding whether the probability that the considered system satisfies the underlying property specified by a temporal logic meets a given threshold or not. A wide range of quantitative performance, reliability, and dependability measures can be specified by means of temporal logics: PCTL for Discrete Time Markov Chains (DTMC) and CSL for Continuous Time Markov Chains (CTMC) [52].

Formal verification by model checking and performance and dependability evaluation have a lot of things in common. The models of real systems are specified by high-level specification techniques as stochastic Petri nets, stochastic process algebra, stochastic automata networks, etc.. Therefore it is possible to construct very large and complex models by using a high-level specification technique. By means of a temporal logic it is possible to express complex state-based and path properties. Thus probabilistic measures over paths as well as standard transient-state and steady-state measures can be specified in a compact and unambiguous manner.

In the last years, the statistical probabilistic model checking approach constitutes an interesting alternative to numerical model checking techniques for large scale systems. The principle of the method is based on stochastic simulation and is quite simple: it generates samples of the behavior i.e. paths and simultaneously check the satisfiability of the formula by the path. Then by a simple count, it estimates (with some confidence level) whether the satisfaction probability is below a low threshold or above a high threshold. The main advantages of this approach are the following:

- we do not generate the state graph, and during generation of a path, only the current state is maintained. This greatly reduces the amount of memory and makes possible to consider infinite-state models.
- the method is applicable for any stochastic process with a well defined semantics. In partic-

ular, no Markovian hypothesis is required.

- Paths initiated from a state can be generated independently, thus parallel simulation techniques can be used.

The verification of the steady-state measures had been undertaken by numerical methods rather than by simulation due to the steady-state detection problem. During CheckBound project, we have shown the efficiency of the verification of steady-state properties by applying perfect-sampling for finite-state Markovian models. First, since perfect sampling provides samples according to the steady-state distribution, it is possible to estimate steady-state properties within required accuracy. Secondly, the functional perfect-sampling is particularly well adapted for the steady-state property checking since we are interested in the probability sum of states satisfying a property. Therefore the steady-state verification of very large scale models can be done by means of perfect sampling [43].

The computational complexity of statistical verification is essentially due to two factors: the sample size (corresponding to the chosen accuracy), on one hand, and the duration of each random experiment (i.e. the length of each random trajectory) generated with respect to the considered discrete-state space, on the other. Obviously, generating random experiences in parallel accelerates potentially the statistical verification. The second problem (duration of each random experience) is especially important for time-unbounded path formulas (probability of execution paths in which some properties are satisfied in time interval $[0, \infty]$).

Objectives of the task. We consider to explore two techniques in this project to optimize a random path checking:

- Application of time-parallel simulation to check time-unbounded path properties. Combining different path segments corresponding to executions different time intervals will require some precautions for verification purposes. It is interesting to emphasize here that in verification we do not need generally exact values, thus bounds may be sufficient to conclude. In this case we have three-valued result (1-satisfied, 2- not satisfied 3- can not be decided) instead of two-valued checking (1-satisfied, 2- not-satisfied).

- Application of numerical methods to accelerate random path (trajectory) generation. Path generation will proceed as long as a satisfying or contradicting state will be reached. However, since depending on the considered property the state-space may be partitioned in a number of Strongly Bottom Connected Component (SBCC), it may be possible that when the generated trajectory enters a certain SBCC then some quantitative information (obtained by application of numerical techniques on the component), may be exploited in order to halt the experiment (i.e. stop the generation of the trajectory) reducing, in such way, its duration.

An important advantage of traditional model checking is its ability to provide diagnostic information as well as rigorous correctness proofs. A counter-example is a failure path for which the desired property is violated. It is particularly useful to locate the cause of a property violation. In the stochastic model checking case, counter-examples are not provided since they are formed by a set of paths. This set must be small to be meaningful as debugging information, thus we look for a small set of paths with high execution probabilities. Parallel path generations by using also some numerical techniques may lead us to generate meaningful counter-examples.

3.3.8. WP5, Task 5.4: Applications to Laser Physics and Statistical Mechanics

Context. Starting from the 2000s, J. Arnaud, L. Chusseau and F. Philippe began to work on the Monte Carlo simulation of lasers [18–20, 40]. The initial motivation of the study was to understand the origin of the optical noise in semiconductor lasers. An original vision of the laser based on Markov process has been implemented for this aim [18]: the operation of the semiconductor laser is completely described by the temporal evolution of carriers in the particular component of their interactions with each other and with the population of photons stored in the laser cavity. Optical noise comes then from random creation or annihilation times of photons and the statistics of the noise intensity derives straightforwardly. Among its initial objectives, this study was intended to understand how it is possible to emit laser light with a sub-Poisson statistics, a feature intrinsically linked to the quantum nature of physical processes involved [21]. Our results were then completely similar to previous ones [94] and showed that the optical noise of a single-mode laser reproduces the electrical noise injection at low frequency as the pump remains low and does not induce too strong nonlinear processes [4–6]. Beyond, the statistics diverge and the laser becomes noisier. In recent years we have applied the same method to lasers with multiple optical modes. The goal this time is to model numerically the intrinsic dynamics of the laser and to understand what may allow a laser to operate simultaneously on multiple modes (namely two) [1, 2, 17]. An application that one may consider (and funded by the ANR PNANO "BASTET") is the ability to generate a ultra-compact terahertz-source from the stable beating of two modes. Our vision of this problem still uses Markov processes and allows to precisely define the limits of stable two-mode behavior. As a consequence the technological manufacturing has been directed toward quantum dots instead than quantum well for the active part of the laser [39]. This study is still in progress and should soon lead to publications in peer-review journals. There is no equivalent model using Markov processes in the scientific literature. Traditional models are either quantum [94], or can reasonably describe a multimode laser by sets of coupled differential equations solved numerically [67, 93]. The only model approaching ours involves cellular automata, but it raises many questions about the underlying physics, especially because it assumes a localization of photons identical to that of electrons, which is totally unreasonable if one compares their respective masses.

Objectives of the task. In this project we want to continue to implement our Monte Carlo modeling of multimode lasers to study the intensity noise and modal noise for a two-mode laser because this feature will dictate the final noise performance of the corresponding terahertz source. We plan to confront our model to the approaches of WP 1 and WP 2, thereby providing a benchmark for the techniques developed there.

3.3.9. WP5, Task 5.5: Markovian Games and Economic Problems

Context. Stochastic games or Markovian games are a special case of stochastic dynamical games, in which the game is played during several stages and in which the moves of the game and the rewards of the players are stochastic and depend only on the current state of the system as well as the joint actions of all the players [83]. The rewards are accumulated along the different stages to give the utility that is optimized. In this context, each player finds its "best" strategy (the rule that determines its actions) owing to the strategies of the other players. This is done by finding an (Nash) equilibrium of the game. Once the strategies are determined, the dynamical behavior of the system is described by a Markov chain. The main numerical problems are twofold. The implementation of efficient methods that computes an equilibrium of the game is a key point. Particularly, because the curse of dimensionality limits the study to games with few players. The

current algorithms are decomposed in two main steps. A temporal step related to the computation of the value in the different states and a game-theoretic step related to the determination of the equilibria. Two main approaches are used. The first one uses the value iteration algorithm coupled with standard approaches of game theory (either convex analysis or cobweb computations) [86]. The other one uses learning approaches coupled with standard methods of game theory or fictitious play [66]. On the other hand, it should be useful to determine a value of the game even when players do not use their Nash strategy (but problems of convergence could appear). Such a computation is based on the stationary distribution of a Markov chain.

Objectives of the task. Our aim is to implement algorithms that compute equilibria assuming perfect information and Markov (or stationary) strategies. Up to our knowledge, there does not exist any integrated framework or software that allows to compute easily any Perfect Markov Equilibrium. For instance, the modeling environment GAMS (largely used in the economic community) does not offer the possibility to compute equilibria. This is why we want to implement several algorithms that compute efficiently one of the equilibria of a game and include them in an environment to facilitate the handling by non computer scientists. Stochastic games and Markov strategies are widely used in Industrial Organization works. They also model the resource sharing of rare resources. We plan to use the programs developed before in common works with economic researchers of University Paris Ouest Nanterre and among them professors J. Etner and P.-A. Jouvet. The topics of the studies will be dedicated to the sharing of pollution permits [73] and the investments required to limit pollution in the field of international climate agreements [54]. At last, we will consider that agents have preferences (avoid risky actions e.g. [62]). This restricts the strategy set and can lead to systems without equilibrium. In this case the most interesting and promising way is to exhibit the dynamical behavior of the system and the value of the game. No software exists now for this and we plan to develop one, based on the results on Markov Chains developed in WP 3 and WP 4.

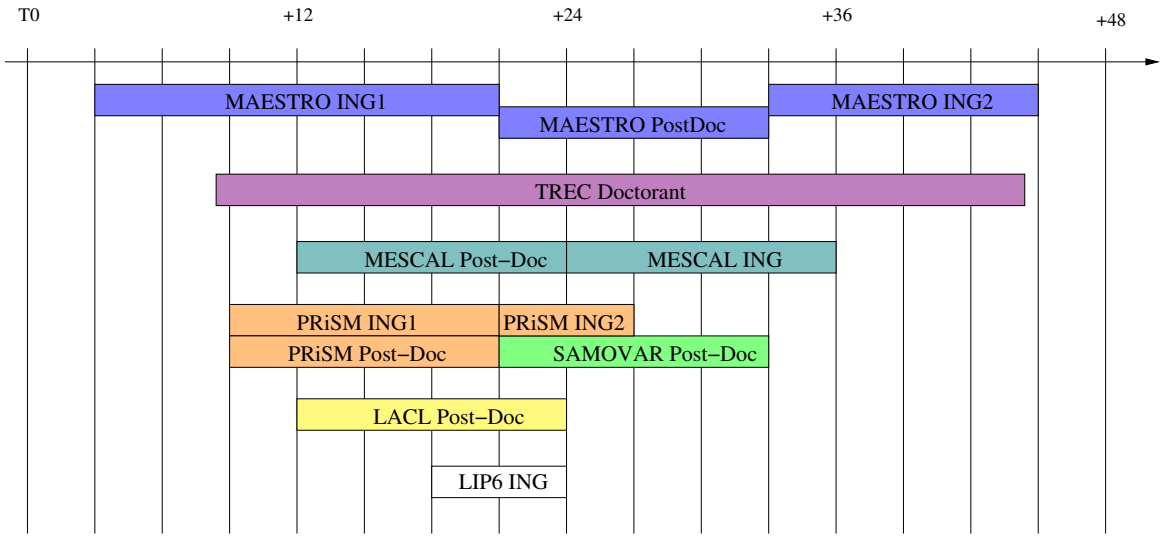
3.4. TASKS SCHEDULE, DELIVERABLES AND MILESTONES

There will be a continuous activity on most tasks during the project, through the involvement of permanent researchers. The scheduling of tasks is better visualized with the activity of the temporary personnel, see the Gantt diagram below. Salient features are:

- The work on software will start after the partners have exchanged among themselves and prepared more detailed specifications; a kickoff meeting will be organized between T0 and T0+1 for this purpose; the preparation work of the engineer of WP 3 will be validated by all partners at T0+3; the development work will start then. A functional version should be delivered at T0+21, with beta versions along the way.

- After T0+21, the research work on numerical algorithms starts in WP 4. When it is completed at T0+33, a second engineer is hired to complete the implementation in the environment, help application task partners and generally maintain the platform. The final version of the environment will be delivered at T0+45.

- The PhD candidate on WP 1 will be typically hired at T0+8, preferably right after a Master's internship.



- The work of the INRIA/Mescal engineer on WP 1 will follow the fundamental investigations of the post-doc on the same subject and will also be coordinated with the work of other engineers (typically with the INRIA/Maestro engineer who will build the software environment).
- The recruitment of engineers for the application tasks occurs after the modeling work has been completed, and also after the environment has been specified and prototyped.

4. DISSEMINATION AND EXPLOITATION OF RESULTS. INTELLECTUAL PROPERTY

The dissemination of our results and software development effort will consider several dimensions:

- Our publication strategy will include high-visibility journals and conferences in applied probabilities and performance modeling. In particular, we shall present our modeling environment in "tool sessions" of conferences and workshops. The "multidisciplinary" approach to applications will be validated through publications in the literature of the application domain.
- A web site will be developed for gathering the software development; its purpose will be in particular to foster the use by scientist outside the MARMOTE project. This web access to the tool is essential so that potential users can gain familiarity with a minimum installation and learning effort.
- We shall use our reputation and links with researchers of the community of Performance Modeling worldwide (*e.g.* the IFIP WG 7.3) to foster the contribution to the software environment. The architecture chosen (see WP 3) and the software licencing should make this as easy as possible. We foresee that the bulk of the project will be available under some

Open Source licence (probably a CECILL licence), while specific modules/methods might be kept under more stringent control by their contributors.

- We shall foster the use of the software in teaching among the participants of the project. This effort will be extended at the international level, for instance through the EIT ICT Labs, of which INRIA and UVSQ partners are members.
- We shall dedicate effort to recruiting new application users in different disciplines. The partners of this project have contacts in Biology, Economics, Geography, left "unexploited" in this proposal, the scientific project not being mature enough, or for temporary lack of availability on the part of prospective partners.
- The project has also a strong potential of industrial applications, as demonstrated by the interest of Alcatel in Task 5.1. Strengthening these links will one of the goals of the management of the project.

Finally, we will seek to make the software realizations "live" beyond the time frame of the project. Depending on the success obtained in recruiting contributors and users (national and international), a sequel will be proposed to the ANR, or for European funding. The possibility to obtain a long-term support from INRIA for the software platform will be also studied.

5. TEAM DESCRIPTION

5.1. PARTNERS DESCRIPTION & RELEVANCE, COMPLEMENTARITY

The consortium is formed of researchers of two "families". On the one hand, researchers involved in the methodological and software developments ("core" of the project) have an experience of joint projects and have been working with each other (in smaller groups) in the past. They all have a common scientific background, but each partner or group of partners brings a different specialty, thereby covering the scope of the project, with some healthy overlapping guaranteeing the seamless communication within the project. INRIA/MAESTRO (A. Jean-Marie) brings an experience in analytic methods for the solution of Markovian models, and the experience in Modeling environments (participation to the ESPRIT II project IMSE and the Modline environment; realization of the ERS environment on (max,+)-related methods). He also has a broad experience of modeling techniques and tools for dynamical systems, control theory, game theory. INRIA/Mescal (J.-M. Vincent and B. Gaujal) brings an experience on Perfect Simulation, materialized in the software package Psi2. INRIA/Trec brings its experience on perfect sampling and bounding methods for Markov chains. INRIA/Mescal and UVSQ/PRiSM bring their experience in Parallel Simulation and algebraic methods for the numerical solution of Markov Chains, materialized in the software package PEPS. UPEC/LACL bring their experience in numerical bounding methods and their applications for model checking. Nihal Pekergin has participated to the development of a statistical model checking tool COSMOS, and to the development of Xborne tool when she was with PRiSM.

Participants are well positioned within the scientific community: A. Jean-Marie and J.-M. Fourneau are elected members of the IFIP WG 7.3, a working group dedicated to stochastic modeling. B. Gaujal and A. Jean-Marie are TPC chairs of the ValueTools'2012 conference, dedicated to Performance Evaluation Methodologies and Tool.

On the other hand, researchers involved in the application tasks are all involved in the application of Markov chain modeling to various fields of science, in addition to theoretical developments. They have developed models, software and algorithms for this. They also have worked at some point with some member of the core of the project and they are interested in applying the technical developments realized within this project, to their own field of research. UPMC/LIP6 brings their expertise in optimization and their links with Economists. TSP/SAMOVAR brings their connections with industrial partners, in particular Alcatel, and their experience in modeling networking applications. Researchers of LIRMM and IES laboratories, associated with the INRIA/Maestro partner, bring their long-time expertise in Markov modeling and simulation of lasers.

5.2. QUALIFICATION OF THE PROJECT COORDINATOR

Alain Jean-Marie, Senior Research scientist at INRIA, has been involved in collaborative projects for a long time. He has worked on the european projects ESPRIT II IMSE, ESPRIT III PEPS, and QMIPS, for which he was partner head for INRIA. He has led several industrial research contracts with Thomson, EDF, ESA and many startup companies. He also served as partner head in the ANR Masses de Données FLUX (2004-2007) and to the ANR Multimedia VOODDO (2008-2011).

The coordinator will be backed by the Contract and Finances service of INRIA Sophia-Antipolis Méditerranée, and its wide experience of collaborative contracts: ANR, Europe or Industrial. He will also count on the participation of Jean-Michel Fourneau, head of the team EPRI in the PRiSM laboratory. He has lead the ACI Sécurité Informatique SR2II project, participated to several ANR projects and been involved in the EURO-NGI/EURO-FGI excellence networks.

5.3. QUALIFICATION AND CONTRIBUTION OF EACH PARTNER

Partner 1: INRIA/MAESTRO				
Jean-Marie	Alain	DR INRIA	36	Project leader, partner head, co-responsible for WP 3 and WP 4, responsible for Task 5.4

Partner INRIA/Maestro (Alain Jean-Marie) brings its experience and scientific network in various aspects of Markov modeling. It will participate to all methodological work packages and application tasks 5.4 and 5.5. For the purpose of this project, L. Chusseau (CNRS/IES Laboratory) and F. Phillipe (Univ. Montpellier 2/LIRMM) are associated with this partner to work on Task 5.4; their active participation is not explicitly accounted due to administrative complications.

Partner 2: INRIA/TREC				
Busic	Ana	CR INRIA	24	Partner head, co-responsible for WP 1, participant to WP 2, WP 4, Task 5.3 and Task 5.5
Bouillard	Anne	MCF	12	participant to WP 1 and Task 5.5

Partner INRIA/Trec (Ana Busic and Anne Bouillard) brings its experience in Markov Chain analysis (perfect sampling, stochastic comparison, bounding chains and non-monotone analysis). It will mainly participate in the methodological work packages WP 1, WP 2, WP 4, and also to some application tasks packages Task 5.3 and Task 5.5.

Partner 3: INRIA/MESCAL				
Gaujal	Bruno	DR INRIA	10	Partner head, co-responsible for WP 1, participant to Task 5.5
Vincent	Jean-Marc	MCF	12	participant to WP 1 and WP 2
Clévenot-Perronnin	Florence	MCF	10	participant to WP 1

Partner INRIA/Mescal brings its expertise in Markov chain analysis, discrete event stochastic systems and dynamics control and games. It will mainly participate in the methodological work package WP 1 with marginal implication in Task 5.5.

Partner 4: UVSQ/PRiSM				
Fourneau	Jean Michel	PR	32	Partner head, responsible for WP 2, co-responsible for WP 3, Task 5.1, Task 5.3
Thu Ha	Dao Thi	CR CNRS	5	Participant to WP 2 and WP 4
Quessette	Franck	MCF	18	Co-responsible for Task 5.2
Vial	Sandrine	MCF	12	Co-responsible for Task 5.2

PRiSM is the computer science laboratory of University of Versailles St Quentin. The University is an active member of TERATEC and RTRA Digiteo. It is also a CNRS UMR (8144). The researchers are members of the performance evaluation team (EPRI) and the Algorithmic team (ALCAaP). They have already conducted many collaborative researches either on theoretical topics (stochastic comparisons, algorithms) or on applications (optical networks, routing, mobile networks, biological models) as it can be checked with the publication lists on the website of the groups. These activities have been sponsored by ANR, DIGITEO or System@tic. They are members of Labex Digiworld (Computer Science) and Charmatte (Chemistry, for an action on discrete modeling of molecules). The four members of the project bring various skills to the project, from theoretical computer science (Dao Thi), applied probability (Dao Thi, Fourneau), numerical algorithms (Fourneau, Quessette), biologie (Quessette, Vial), Parallel Simulation (Fourneau, Quessette, Dao Thi).

Partner 5: Telecom SudParis/SAMOVAR				
Castel-Taleb	Hind	MCF	20	Partner head, co-responsible for Task 5.1, participant to WP 1 and WP 2
Lourdiane	Mounia	MCF	8	Participant to Task 5.1
Jakubowicz	Jérémie	MCF	8	Participant to WP 2 and Task 5.1

Partner Telecom SudParis/SAMOVAR has a good experience both on theoretical aspects related to methods of performance evaluation and applied analysis of network architectures. Hind Castel-Taleb has worked on stochastic bound methods for analysis of large state system. She has worked also on traffic analysis in optical networks, and the definition of aggregation mechanisms. Mounia Lourdiane works on architecture and dimensioning problems in communication networks. Mounia Lourdiane and Hind Castel-Taleb are working actually on performance evaluation of 100Gbit optical networks in 100GRIA SYSTEMATIC project (June 2010-August 2012). Jérémie Jakubowicz is mainly involved in research on distributed algorithms. This covers modeling, algorithm design, and performance evaluation of such algorithms.

Partner 6: UPEC/LACL				
Pekergin	Nihal	PR	18	Partner head, co-responsible for Task 5.3
Tran	Minh-Anh	MCF	8	Participant to WP 2, WP 4, Task 5.3
Tan	Sovanna	MCF	8	Participant to WP 3, Task 5.3

LACL is the Computer Science laboratory of University Paris-Est Créteil. The three members of the project bring various skills to the project. Sovanna Tan has a large experience in parallel programming, Minh-Anh Tan in applied probability, Nihal Pekergin in quantitative verification, and bounding techniques.

Partner 7: UPMC/LIP6				
Hyon	Emmanuel	MCF	14	Partner head, responsible for Task 5.5, participant to WP 3 and WP 4
Delbot	François	MCF	5	Participant to Task 5.5, WP 3 and WP 4

LIP6 brings an experience in computing economic figures of dynamical systems by numerical simulations during the collaboration with European Spatial Agency (ESA). E. Hyon collaborates with Economists in the project ALGECO funded by the CNRS GDR RO. F. Delbot is mainly involved in research on average analysis of algorithms and algorithm design.

6. SCIENTIFIC JUSTIFICATION OF REQUESTED RESOURCES

The distribution of manpower per task/work package and per partner is detailed in a table in the Annex. The following description is limited to temporary staff.

6.1. PARTENAIRE 1 / PARTNER 1 : INRIA EQUIPE-PROJET MAESTRO

- Équipement / Equipment

Computer equipment for permanent and temporary staff: 2 server/workstation and 2 laptops.

- Personnel / Staff

30 man x months for software development engineers: 2 different persons (one for software architecture, beginning of the project; one for numerical methods, second part of the project);

12 months for a Post-doctoral position. Role: development of theory and algorithms for Markov-Additive processes.

- Prestation de service externe / Subcontracting: None.

- Missions / Travel

5500 Euros/year to organize the following travels: -general assemblies of the project (2/year, total 1 kE/year); -developer meetings, code sprints etc (*a priori* for the engineering staff); -visits or invitations of teams outside the project, potential contributers to the environment (WP 3) or potential users of simulations techniques (WP 1, WP 2or WP 4) (4 in total, 1.5kE/year average); -conferences (3 kE/year).

• Facturation interne / Costs justified by internal procedures of invoicing

INRIA internal cost for project management amounts to 4% of the costs specified above.

• Autres dépenses de fonctionnement / Other expenses: None.

6.2. PARTENAIRE 2 / PARTNER 2 : INRIA EQUIPE-PROJET TREC

• Équipement / Equipment

Computer equipment for permanent and temporary staff: 1 server/workstation and 1 laptop.

• Personnel / Staff

One PhD student (36 months): 116 379 Eur. One intern, Master 2 (6 months): 12 576 Eur.

PhD student job description

The thesis will be focused on WP 1(Section 3.3.1), sub-tasks 1.1. and 1.2. The models considered will be mainly random graphs and other combinatorial structures. The candidate will consider both algorithmic and theoretical aspects of perfect sampling. For the algorithmic part, the main difficulty is to adapt envelope computation or other coupling detection methods to considered models. The theoretical part will include the study of the coupling time.

• Prestation de service externe / Subcontracting: None.

• Missions / Travel

5 500 euros per year to organize the following travels for the WP 1, collaborative works and presentation of the results obtained by the team: - Missions for general assemblies of the project; - Missions for visiting teams for discussions about methodology, algorithms, implementations and applications (many visits to INRIA/Mescal for the PhD student); - Missions abroad to present results at international conferences. The total travel expenses amount to 22 000 Euros.

• Facturation interne / Costs justified by internal procedures of invoicing

INRIA internal cost for project management amounts to 4% of the costs specified above.

• Autres dépenses de fonctionnement / Other expenses: None.

6.3. PARTENAIRE 3 / PARTNER 3 : INRIA EQUIPE-PROJET MESCAL

• Équipement / Equipment

Two laptops, for the equipment of non-permanent staffs (3000 Euros)

• Personnel / Staff

One engineer (12 months) and one postdoc (12 months).

Engineer job description

The engineer will be responsible of the design and the construction of the new perfect simulator. This tool, described in Task 3.3.1, will constitute the main output of the global Task 3.3.1: It will implement all the algorithmic and structural results of the first work package. The tool will allow for advanced data structures enabling amortized costs, in particular for skipping useless events in the simulation. It will also encompass semi-automatic parallelization.

Post-doc job description

The postdoc researcher will mainly work on WP 1(Section 3.3.1). His/her work will include the analysis of the coupling time of the Markov chain for several models that are not monotone, such as networks of queues, Petri nets or random structures. S/He will also investigate the optimal splitting time of the perfect sampling algorithm and the computation of envelopes for the same kinds of models. Finally, s/he should also be involved with one or two applications described in Task 5.1, Task 5.2 or Task 5.4, namely networks dimensioning, biological models or statistical mechanics computations.

• Prestation de service externe / Subcontracting: None.

• Missions / Travel

4 persons participating to 3 meetings per year ($300 \times 12 = 3600$ Euros/year) and three conferences per year ($1500 \times 3 = 4500$ Euros/year), one long term visit to another partner for a non-permanent engineer (500 Euros/year), the total is 8600 Euros/year. The total travel expenses amount to 34400 Euros.

• Facturation interne / Costs justified by internal procedures of invoicing:

INRIA internal cost for project management amounts to 4% of the costs specified above.

• Autres dépenses de fonctionnement / Other expenses: None.

6.4. PARTENAIRE 4 / PARTNER 4: PRiSM, UNIVERSITÉ DE VERSAILLES ST QUENTIN

• Équipement / Equipment

None. We do not need any equipment. Indeed, the team has already bought a multi-core workstation with a Tesla GPU card which more than 440 cores. This machine is sufficient to experiment with the use of a GPU for time-parallel simulation. The GPU are operated with multiple simple cores running the same instruction on different data. For GRID computing, the team expects to have access to GRID 5000 infrastructure due to the collaboration with INRIA teams (Mescal). Stations and laptops are already available for non permanent staff.

• Personnel / Staff

- 1 year of post doc to help the development of the theory of qualitative properties and how they can speed up the time-parallel simulations and the space decomposition approaches as well.
- 12 months for an engineer for software development of the parallel simulation of the model of apoptosis network in a cell (note that the software development has already begun and it

was paid by our own resources).

- 6 months for an engineer for the development of the two time- parallel simulation approaches on GPU.

• Prestation de service externe / Subcontracting

2000 euros per year. We plan to work with EGIDE to pay some Master students for some research activities during an internship to complete their second year of master. These internships will help us to find young talented researchers who can then join the teams as PhD students after applying on a scholarship from one partner.

• Missions / Travel

6000 euros per year for laboratory fees and to organize the following travels for the WP1, collaborative works and presentation of the results obtained by the teams: -Missions for general assemblies of the project (2 per year); - Missions for visiting teams for discussions about methodology, algorithms, implementations and applications (not very expensive because many teams are in the suburbs of Paris); -Missions abroad to present results at international conferences (3 per year)

• Facturation interne / Costs justified by internal procedures of invoicing

The University of Versailles Saint Quentin deducts in general 10% on research contracts but this deduction is limited to 4% to follow the rules of ANR.

• Autres dépenses de fonctionnement / Other expenses: None.

6.5. PARTENAIRE 5 / PARTNER 5 : SAMOVAR – TELECOM SUD-PARIS

• Équipement / Equipment

Computer equipment for temporary staff : 1 server/workstation.

• Personnel / Staff

1 postDoc for 12 months : research theory for introducing in simulation (parallel and perfect) methods specific constraints of cloud computing. We plan to add to the tools implementing the models (to be done by the research masters) the functionalities to deploy the cloud environment on the simulators developed by other partners. This tool will be used to test automatically cloud environments given by Alcatel Lucent in order to propose dimensioning solutions.

• Prestation de service externe / Subcontracting

2 Research Masters (M2): definition of models and development of tools in order to implement cloud environment and network architectures given by Alcatel Lucent. For one research master, the cost is 5000 euros (1000 euros per month for 5 months by Egide).

• Missions / Travel

4000 euros per year for project meetings and travel to conferences in order to present papers related to the project.

• Facturation interne / Costs justified by internal procedures of invoicing

TSP takes 5% of research contracts.

• Autres dépenses de fonctionnement / Other expenses: None.

6.6. PARTENAIRE 6 / PARTNER 6 : LACL – UNIVERSITÉ PARIS-EST CRÉTEIL

• Équipement / Equipment: None.

• Personnel / Staff

One year post-doc researcher. The focus of his/her work will be primarily on the research lines outlined in Task 5.3, Stochastic Model Checking by Simulation. S/He will be in contact with other partners in order to take advantage of the theoretical results of the project and also to use the tools and environments that will be developed. The post-doc researcher will work on the efficient path generation for time-unbounded properties that may occur in time interval $[0, \infty]$ by exploring how one can combine numerical methods and statistical techniques and how time-parallel simulation can be applied to accelerate sample generation. The post-doc researcher will also investigate on the meaningful counter-example generation for the stochastic model checking case.

• Prestation de service externe / Subcontracting

We plan to work with Egide to pay two Master students over the duration of the project for their internships to complete their second year of Master. Cost: 6000 Euros.

• Missions / Travel

4000 Euros per year to organize the following travels: -Missions for general assemblies and for visiting other partners for working meetings; -Missions to present project-related papers in international conferences.

Facturation interne / Costs justified by internal procedures of invoicing

UPEC deducts 4% on research contracts.

• Autres dépenses de fonctionnement / Other expenses

Computer equipment for temporary staff: 1 server/workstation, 1 laptop. Cost: 4000 Euros.

6.7. PARTENAIRE 7 / PARTNER 7 : LIP6 – UNIVERSITÉ PIERRE-ET-MARIE-CURIE

• Équipement / Equipment

Computer equipment for temporary staff: one laptop.

• Personnel / Staff

One engineer during 6 months. Role: implement different algorithms related to the computation of Markov game equilibria in the same programming language than the one used for the

environments tools; implement the common dynamic programming algorithms used both in game theory and MDP; adapt and implement an easy-to-use interface for non computer scientists. Cost: 27 360 Euros.

Master Trainees: One trainee focused on the use of numerical simulations for economic problems in which agents have preferences. One trainee focused on the use Markov Games in climate agreement problems. Cost: 6000 Euros (3000 Euros for a 6 month M2 trainee).

- Prestation de service externe / Subcontracting: None.
- Missions / Travel

2000 Euros per year to participate in project meetings and to travel in order to present the results of the project in conferences.

- Facturation interne / Costs justified by internal procedures of invoicing

The University Pierre-et-Marie-Curie deducts 4% on contracts.

- Autres dépenses de fonctionnement / Other expenses: None.

7. REFERENCES

REFERENCES OF PARTNERS

[1] A. Akwoué Ondo, J. Torres, P. Nouvel, C. Palermo, L. Chusseau, J. Jacquet, and M. Thual. Analysis of the stable two-mode operation of a 4-sections semiconductor laser for THz generation by photomixing. In *International Symposium on Space Terahertz Technology*, pages 255–258, Paris, May 2006.

[2] A. Akwoué Ondo, J. Torres, C. Palermo, L. Chusseau, J. Jacquet, and M. Thual. Two-mode operation of 4-sections semiconductor laser for THz generation by photomixing. In D. Jäger and A. Stöhr, editors, *Proceedings of SPIE: "Millimeter-Wave and Terahertz Photonics"*, volume 6194, pages 164–170, Strasbourg, April 2006.

[3] Jonatha Anselmi and Bruno Gaujal. On the efficiency of perfect simulation in monotone queueing networks. In *IFIP Performance: 29th International Symposium on Computer Performance, Modeling, Measurements and Evaluation*, Amsterdam, October 2011. ACM Performance Evaluation Review.

[4] J. Arnaud, L. Chusseau, and F. Philippe. Sub-Poissonian laser emission from a single-electron permanently interacting with a single-mode cavity, 2007. arXiv:0710.2596.

[5] J. Arnaud, L. Chusseau, and F. Philippe. Quiet lasers, 2009. arXiv:0906.1469.

[6] Jacques Arnaud, Laurent Chusseau, and Fabrice Philippe. Grand-mother clocks and quiet lasers, 2009. arXiv:0901.0983.

[7] François Baccelli, Bartłomiej Błaszczyszyn, and Florent Tournois. Spatial averages of coverage characteristics in large CDMA networks. *Wirel. Netw.*, 8:569–586, 2002.

[8] François Baccelli, Bruno Gaujal, and Nathalie Furmento. Parallel and Distributed Simulation of Free Choice Petri Nets. In *Parallel And Distributed Simulation*, Lake Placid, USA, 1995.

[9] Anne Benoit, Leonardo Brenner, Paulo Fernandes, Brigitte Plateau, and William J. Stewart. The peps software tool. In Peter Kemper and William H. Sanders, editors, *Computer Performance Evaluations,*

Modelling Techniques and Tools. 13th International Conference, TOOLS 2003, Urbana, IL, USA, September 2-5, 2003, volume 2794 of *Lecture Notes in Computer Science*, pages 98–115. Springer, 2003.

[10] E. Biersack, A. Jean-Marie, and Ph. Nain. Open-loop video distribution with support of vcr functionality. In *Performance'02*, sep 2002. Performance Evaluation, 49 (1/4): 411–427.

[11] Anne Bouillard and Bruno Gaujal. Backward coupling in bounded free-choice nets under Markovian and non-Markovian assumptions. *Journal of Discrete Event Dynamic Systems (JDEDS)*, 18(4):473–498, 2008.

[12] A. Bušić, J. Mairesse, and I. Marcovici. Probabilistic cellular automata, invariant measures, and perfect sampling. In *Proc. of 28th International Symposium on Theoretical Aspects of Computer Science, (STACS)*, pages 296–307, 2011.

[13] A. Bušić and J.-M. Fourneau. Iterative component-wise bounds for the steady-state distribution of a Markov chain. *Numerical Linear Algebra with Applications*, 18(6):1031–1049, 2011.

[14] Ana Bušić, Bruno Gaujal, Gaël Gorgo, and Jean-Marc Vincent. PSI2 : Envelope Perfect Sampling of Non Monotone Systems. In *International Conference on Quantitative Evaluation of Systems (QEST)*, pages 83–84, Williamsburg, Virginia, USA, 2010. IEEE Computer Society.

[15] Ana Bušić, Bruno Gaujal, and Furcy Pin. Perfect sampling of Markov chains with piecewise homogeneous events. *Performance Evaluation*, 2011. accepted for publication.

[16] Ana Bušić, Bruno Gaujal, and Jean-Marc Vincent. Perfect simulation and non-monotone Markovian systems. In *3rd International Conference Valuetools'08*, Athens, Greece, October 2008. ICST.

[17] L. Chusseau, A. Akwoué Ondo, J. Torres, P. Nouvel, J. Jacquet, and M. Thual. Four-sections semiconductor two-mode laser for THz generation. In P. Mathieu, editor, *Proceedings of SPIE: "Photonics North"*, volume 6343, pages 1097–1106, Québec, June 2006.

[18] L. Chusseau and J. Arnaud. Monte-Carlo simulation of laser diodes sub-poissonian light generation. *Opt. Quant. Electron.*, 34:1007–1023, 2002.

[19] L. Chusseau, J. Arnaud, and F. Philippe. Rate-equation approach to atomic laser light statistics. *Phys. Rev. A*, 66:053818, 2002.

[20] L. Chusseau, J. Arnaud, and F. Philippe. Rate-equation approach to laser light statistics. In *IX^e International Conference on Quantum Optics*, Minsk, 2002.

[21] L. Chusseau, J. Arnaud, and F. Philippe. Étude Monte Carlo de la statistique des photons d'un laser proche du seuil. In *8^e Conférence sur les Lasers et l'Optique Quantique*, volume 119, pages 155–156, Toulouse, 2004. J. Phys. IV France.

[22] Thu-Ha Dao-Thi, J.M. Fourneau, and I. Kadi. Monotone time parallel simulation of a web server. In *IEEE Symposium on Computers and Communications, Pediswesa Workshop*, pages 1097–1102. IEEE Computer Society, 2010.

[23] Jantien Dopper, Bruno Gaujal, and Jean-Marc Vincent. Bounds for the coupling time in queueing networks perfect simulation. In *Numerical Solutions for Markov Chains (NSMC'06)*, pages 117–136, Charleston, 2006. The 2006 A.A. Markov Anniversary Meeting (MAM 2006).

[24] J. Dutheil, T. Pupko, A. Jean-Marie, and N. Galtier. A model-based aproach for detecting co-evolving positions in a molecule. *Mol. Bio. Evol.*, 22(9):1919–1928, sep 2005. [doi:10.1093/molbev/msi183].

[25] J.-M. Fourneau, M. Le Coz, N. Pekergin, and F. Quessette. An open tool to compute stochastic bounds on steady-state distributions and rewards. In *11th International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 2003), Orlando, FL*. IEEE Computer Society, 2003.

[26] J.-M. Fourneau and I. Kadi. Time parallel simulation of monotone systems. Poster Session, IFIP Performance conference, Namur, Belgique, 2010.

[27] J.-M. Fourneau, I. Kadi, and N. Pekergin. Improving time parallel simulation for monotone systems. In S. J. Turner, D. Roberts, W. Cai, and Ab. El-Saddik, editors, *13th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications, Singapore*, pages 231–234. IEEE Computer Society, 2009.

[28] J.-M. Fourneau, I. Kadi, and F. Quessette. Time parallel simulation and hv-monotonicity. In Erol Gelenbe, Ricardo Lent, and Georgia Sakellari, editors, *Proceedings of the 26th Internation Symposium on Computer and Information Sciences*, pages 201–207. Springer Lectures Notes in Electrical Engineering, 2011.

[29] J.-M. Fourneau and N. Pekergin. An algorithmic approach to stochastic bounds. In *Performance Evaluation of Complex Systems: Techniques and Tools, Performance 2002, Tutorial Lectures*, volume 2459 of *LNCS*, pages 64–88. Springer, 2002.

[30] J.-M. Fourneau and F. Quessette. Monotone queueing networks and time parallel simulation. In Khalid Al-Begain, Simonetta Balsamo, Dieter Fiems, and Andrea Marin, editors, *Analytical and Stochastic Modeling Techniques and Applications - 18th International Conference, ASMTA 2011, Venice, Italy*, volume 6751 of *Lecture Notes in Computer Science*, pages 204–218. Springer, 2011.

[31] J.-M. Fourneau and F. Quessette. Monotonicity and efficient computation of bounds with time parallel simulation. In N. Thomas, editor, *8th European Performance Engineering Workshop, EPEW 2011, Borrowdale, The English Lake District*, volume 6977 of *Lecture Notes in Computer Science*. Springer, 2011.

[32] Jean-Michel Fourneau, Imène Y. Kadi, Nihal Pekergin, Jérôme Vienne, and Jean-Marc Vincent. Perfect simulation and monotone stochastic bounds. In Peter W. Glynn, editor, *2nd International Conference on Performance Evaluation Methodolgies and Tools, Valuetools'07, Nantes, France*, Nantes, France, 2007. ICST.

[33] Nicolas Galtier, Olivier Gascuel, and Alain Jean-Marie. Introduction to Markov models in molecular evolution. In Rasmus Nielsen, editor, *Statistical Methods in Molecular Evolution*. Springer, 2005.

[34] Bruno Gaujal, Gael Gorgo, and Jean-Marc Vincent. Perfect sampling of phase-type servers using bounding envelopes. In Springer-Verlag, editor, *18th International Conference on Analytical and Stochastic Modelling Techniques and Applications (ASMTA'11)*, LNCS, Venise, 2011.

[35] Bruno Gaujal, Albert G. Greenberg, and David M. Nicol. A sweep algorithm for massively parallel simulation of circuit-switched networks. *J. Parallel Distrib. Comput.*, 18(4):484–500, 1993.

[36] Bruno Gaujal, Stefan Haar, and Jean Mairesse. Blocking a Transition in a Free Choice Net, and what it tells about its throughput. *Journal of Computer and System Sciences*, 66(3):515–548, 2003.

[37] Bruno Gaujal, Florence Perronnin, and Rémi Bertin. Perfect simulation of a class of stochastic hybrid systems with an application to peer to peer systems. *Journal of Discrete Event Dynamic Systems*, 2007. Special Issue on Hybrid Systems.

[38] A. Jean-Marie. Ers – a software tool for discrete event systems. Demonstration in the Tools Session of WODES2000, 5th Workshop on Discrete Event Systems, Ghent, August 2000.

[39] K. Kusiaku, X. Letartre, C. Seassal, J.-L. Leclercq, P. Rojo-Romeo, P. Regreny, L. Chusseau, F. Disanto, F. Philippe, and E. Augendre. Dual-wavelength optical micro-resonator for frequency conversion. In *6e Journées Térahertz*, La Grande Motte, 25–27 Mai 2011.

[40] F. Philippe, J. Arnaud, and L. Chusseau. Statistics of non-interacting bosons and fermions in micro-canonical, canonical and grand-canonical ensembles: A survey, 2003. arXiv:math-ph/0211029.

[41] Furcy Pin, Ana Bušić, and Bruno Gaujal. Acceleration of perfect sampling for skipping events. In *Valuetools*, Paris, 2011.

[42] B. Plateau, J.M. Fourneau, and K.H. Lee. PEPS: A package for solving complex Markov models of parallel systems. In *Proceedings of the 4th Int. Conf. on Modeling Techniques and Tools for Computer Performance Evaluation, Majorca, Spain.*, pages 341–360, 1988.

[43] Diana El Rabih, Gaël Gorgo, Nihal Pekergin, and Jean-Marc Vincent. Steady state property verification of very large systems. *Int. Journal of Critical Computer-Based Systems, IJCCBS*, 2(3/4):309–331, 2011.

[44] Jean-Marc Vincent and Corine Marchand. On the exact simulation of functionals of stationary Markov chains. *Linear Algebra and its Applications*, 386:285–310, 2004.

[45] Jean-Marc Vincent and Jérôme Vienne. Psi^2 a software tool for the perfect simulation of finite queueing networks. In *QEST'07*, pages 113–114. IEEE Computer Society, 2007.

GENERAL BIBLIOGRAPHY

[46] Nist cloud computing reference architecture. Technical report, National Institute of Standards and Technology (NIST). http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505.

[47] OpenAlea, software enviroment for plant modeling. http://openalea.gforge.inria.fr/.

[48] J. Abate and W. Whitt. The Fourier-series method for inverting transforms of probability distributions. *Queueing Systems*, 10:5–88, 1992.

[49] Patrick Amar, Gilles Bernot, and Victor Norris. Hsim : a simulation programm to study large assemblies of proteins. *Journal of Biological Physics and Chemistry*, 4:79–84, 2004.

[50] S. Andradóttir and T. J. Ott. Time-segmentation parallel simulation of networks of queues with loss or communication blocking. *ACM Trans. Model. Comput. Simul.*, 5(4):269–305, 1995.

[51] Konstantin E. Avrachenkov and Jean B. Lasserre. The fundamental matrix of singularly perturbed Markov chains. *Adv. in Appl. Probab.*, 31(3):679–697, 1999.

[52] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. Representation and Mind. The MIT Press, 2008.

[53] E.A. Bender. Central and local limit theorems applied to asymptotic enumeration. *J. of Combinatorial Theory*, 15:91–111, 1973.

[54] T. Brechet, F. Gérard, and H. Tulkens. Efficiency vs. stability of climate coalitions: a conceptual and computational appraisal. *The Energy Journal*, 32(1):49–76, 2011.

[55] C. T. Butts. A perfect sampling method for exponential random graph models. slides, 2009.

[56] H. Caswell. *Matrix population models: Construction, analysis and interpretation*. Sinauer Associates, Sunderland, Massachusetts, 2nd edition, 2001.

[57] E. Cinlar. *Introduction to Stochastic Processes*. Prentice Hall Press, 1975.

[58] E. De Souza e Silva. Presentation of the Tangram II modeling environment. Technical report, Federal University of Rio de Janeiro (UFRJ)., http://www.land.ufrj.br/tools/tangram2/tangram2.html, 2012.

[59] Charo I. Del Genio, Hyunju Kim, Zoltán Toroczkai, and Kevin E. Bassler. Efficient and exact sampling of simple graphs with given arbitrary degree sequence. *PLoS ONE*, (4), 2010.

[60] M. Dyer and C. Greenhill. Random walks on combinatorial objects. *Surveys in Combinatorics*, 1999.

[61] Martin Dyer and Catherine Greenhill. On Markov chains for independent sets. *Journal of Algorithms*, 35:17 – 49, 2000.

[62] J. Etner and M. Jeleva. Risk perception, health prevention and diagnostic test. *Health Economics*, 2012. To appear.

[63] Philippe Flajolet and Robert Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009.

[64] R. M. Fujimoto, C. A. Cooper, and I. Nikolaidis. Parallel simulation of statistical multiplexers. *J. of Discrete Event Dynamic Systems*, 5:115–140, 1994.

[65] Richard M. Fujimoto. *Parallel and Distributed Simulation Systems*. Wiley Series on Parallel and Distributed Computing, 2000.

[66] D. Fundenberg and D. Levine. *Theory of Learning in Games*. MIT Press, 1999.

[67] Z. Gao, L. Wang, and J.-J. He. Mode competition analysis in dual-wavelength coupled-cavity semiconductor laser. *J. Opt. Soc. Am. B*, 27(3):432–441, 2010.

[68] S.K. Garg and R. Buyya. Networkcloudsim : Modelling parallel applications in cloud simulations. In *Fourth IEEE International Conference Utility and Cloud Computing*. IEEE Computer Society, 2011.

[69] O. Häggström and K. Nelander. Exact sampling from anti-monotone systems. *Statist. Neerlandica*, 52(3):360–380, 1998.

[70] M. Hlynka. List of queueing theory software. Technical report, Univ. Windsor, http://www2.uwindsor.ca/h̃lynka/qsoft.html, 2011.

[71] D. Huang, W. Chen, P. Mehta, S. Meyn, and A. Surana. Feature selection for neuro-dynamic programming, 2011.

[72] M. Huber. Perfect sampling using bounding chains. *Ann. Appl. Probab.*, 14(2):734–753, 2004.

[73] P.-A. Jouvet, P. Michel, and G. Rotillon. Competitive markets for pollution permits: Impact on factor income and international equilibrium. *Environmental Modeling and Assessment*, 15, 2010.

[74] Wilfrid S Kendall. Perfect simulation for the area-interaction point process. In L Accardi and C C Heyde, editors, *Probability Towards 2000*, pages 218–234, New York, 1998. University of Warwick Department of Statistics, Springer-Verlag.

[75] Wilfrid S Kendall and Jesper Møller. Perfect simulation using dominating processes on ordered spaces, with application to locally stable point processes. *Advances in Applied Probability*, 32(3):844–865, 2000.

[76] V.G. Kulkarni. Effective bandwidth for Markov regenerative sources. *Queueing Systems*, 24:137–153, 1996.

[77] G. Latouche and V. Ramaswami. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. ASA-SIAM, 1999.

[78] Gael Lecellier and Catherine Brenner. Genomic and proteomic screening of apoptosis mitochondria regulators for drug target discovery. *Current Medicinal Chemistry*, 14:1–23, 2007.

[79] David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. *Markov chains and mixing times*. American Mathematical Society, 2006.

[80] T.M. Liggett. *Interacting Particle Systems*. Wiley, 1985.

[81] T.M. Liggett. *Stochastic interacting systems: contact, voter and exclusion processes*. Wiley, 1999.

[82] W. Liu, Z. Du, Y. Chen, X. Chai, and X. Wang. On an automatic simulation environment customizing services for cloud simulation center. In *5th IEEE International Symposium on Service Oriented Engineering*. IEEE Computer Society, 2010.

[83] J.F. Mertens. Stochastic games. In *Handbook of game theory*, chapter 47. North Holland, 2002.

[84] Danil Nemirovsky. Tensor approach to mixed high-order moments of absorbing Markov chains. *Linear Algebra and its Applications*, September 2011. Online version, [doi:10.1016/j.laa.2011.08.027].

[85] M.F. Neuts. The fundamental period of a queue with Markov-modulated arrivals. In *Probability, Statistics and Mathematics: papers in honour of Samuel Karlin*. Academic Press, NY, 1989.

[86] A. Pakes and P. McGuire. Stochastic approximation for dynamic analysis: Markov perfect equilibrium and the 'curse' of dimensionality. *Econometrica*, pages 1261–1281, 2001.

[87] Warren B. Powell. Perspectives of approximate dynamic programming. *Annals of Operations Research*, pages 1–38, 2012.

[88] James G. Propp and David B. Wilson. Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, 9(1-2):223–252, 1996.

[89] H.C. Tijms. *Stochastic Modeling and Analysis – A Computational Approach*. J. Wiley & Sons, 1986.

[90] Luis M. Vaquero, Juan Cáceres, and Daniel Morán. The challenge of service level scalability for the cloud. *International Journal of Cloud Applications and Computing*, March 2011.

[91] D. Verchere. Cloud computing over telecom network. In *Optical Fiber Communication Conference*, March 2011.

[92] D. B. Wilson. Perfectly random sampling with Markov chains. Annotated bibliograpy. `http://dimacs.rutgers.edu/~dbwilson/exact.html/`.

[93] A. M. Yacomotti, L. Furfaro, X. Hachair, F. Pedaci, M. Giudici, J. Tredicce, J. Javaloyes, S. Balle, E. A. Viktorov, and Paul Mandel. Dynamics of multimode semiconductor lasers. *Phys. Rev. A*, 69:053816, 2004.

[94] Y. Yamamoto, S. Machida, and O. Nilsson. Amplitude squeezing in a pump-noise-suppressed laser oscillator. *Phys. Rev. A*, 34:4025–4042, 1986.