Echantillonnage préférentiel pour le model checking statistique

Benoît Barbot — Serge Haddad — Claudine Picaronny

Laboratoire Spécification et Vérification, CNRS & ENS de Cachan
61, avenue du Président Wilson, 94235 CACHAN Cedex, France
barbot@lsv.ens-cachan.fr
haddad@lsv.ens-cachan.fr
picaronny@lsv.ens-cachan.fr

RÉSUMÉ. *Le model checking statistique est une alternative intéressante au model checking numérique lorsque les modèles probabilistes étudiés sont de très grande taille. Cependant l'approche statistique ne permet pas d'évaluer les probabilités des évènements rares. Afin de résoudre ce problème, nous développons ici une nouvelle approche basée sur l'échantillonnage préférentiel. Alors que la plupart des techniques d'échantillonnage préférentiel sont basées sur des heuristiques, nous établissons des résultats théoriques. Moyennant certaines hypothèses, ces résultats garantissent une réduction de la variance lors de l'application de l'échantillonnage préférentiel. Nous caractérisons des situations qui vérifient les hypothèses et étendons notre approche dans les autres situations mais cette fois-ci sans garantie théorique. Nous avons implémenté cette approche à l'aide de l'outil* COSMOS *après avoir ajouté des fonctionnalités. Enfin nous présentons l'évaluation de notre méthode sur deux exemples et analysons les expérimentations.*

ABSTRACT. *The statistical model checking can be usefully substituted for numerical model checking when the models to be studied are huge. However the statistical approach cannot evaluate too small probabilities. In order to solve the problem, we develop here a new approach based on importance sampling. While most of the techniques related to importance sampling are based on heuristics, we establish theoretical results under some hypotheses. These results ensure a reduction of the variance during application of importance sampling. We also characterize situations that fulfill the hypotheses and we extend our approach for handling other situations but then without theoretical guarantee. We have implemented this approach with the tool* COSMOS *after some extensions. At last we have evaluated this approach for two examples and analysed the experimentations.*

MOTS-CLÉS : *model checking statistique, évènements rares , échantillonnage préférentiel*

KEYWORDS: *statistical model checking, rare events, importance sampling*

1. Introduction

Contexte. Le model-checking (Emerson *et al.*, 1980) s'avère une technique efficace et rigoureuse de vérification automatique des propriétés requises pour le fonctionnement correct de systèmes à évènements discrets. Sa simplicité algorithmique a permis le développement de nombreux outils. Au départ restreint à la vérification de propriétés qualitatives, il s'est peu à peu étendu aux aspects quantitatifs nécessaires à l'évaluation des performances des systèmes, en particulier la possibilité de prendre en compte des aspects probabilistes (Kwiatkowska *et al.*, 2007; Baier *et al.*, 2008).

Les techniques numériques pour l'analyse quantitative de systèmes probabilistes restent efficaces et précises mais supposent des hypothèses contraignantes, souvent non vérifiées, sur le système et sur la propriété. Pour dépasser ce cadre restreint, on utilise une approche statistique. On crée par simulation un échantillon suffisamment important de comportements du système afin d'estimer la proportion de comportements vérifiant la propriété requise. Le résultat obtenu n'est qu'un encadrement probabiliste de la quantité recherchée, mais la simplicité algorithmique de la démarche permet de l'utiliser dans un contexte plus vaste (Legay *et al.*, 2010).

Néanmoins, cette approche ne permet pas d'estimer la fréquence des comportements dits rares (c'est à dire ayant une probabilité très faible). Ceux-ci sont quasiment inobservables par simulation. Il est pourtant crucial d'analyser quantitativement de tels comportements lorsque ceux-ci ont des conséquences dramatiques. Pour obtenir tout de même un résultat, deux démarches sont utilisées en statistiques (Rubino *et al.*, 2009), celle de *branchement multi-niveaux* et celle d'*échantillonnage préférentiel*.

Contribution. Nous utilisons la technique d'échantillonnage préférentiel (plus simple et théoriquement plus précise) pour proposer une méthode de model-checking d'évènements rares. Notre méthode consiste à réduire de façon intelligente la chaîne de Markov sous-jacente afin de créer un modèle réduit sur lequel on calcule une distribution de probabilités, à la base d'un échantillonnage préférentiel. A contrario de toutes les autres méthodes, nous donnons un cadre théorique suffisant pour garantir la réduction de la variance et fournir un intervalle de confiance. Notre méthode peut aussi s'appliquer hors de ce cadre mais sans ces garanties. Pour l'expérimenter, nous l'avons implémentée à l'aide de l'outil de model checking statistique COSMOS, en utilisant l'outil PRISM comme outil de calcul numérique. Le modèle en entrée est celui des réseaux de Petri stochastiques et la quantité à évaluer s'exprime comme une formule de la logique HASL (Ballarini *et al.*, 2011). Nous avons testé notre méthode sur deux exemples, le premier, inspiré de la ruine des joueurs, le second sur le modèle du dîner des philosophes. sur ces deux exemples, nous obtenons des résultats hors de portée des outils existants. Nous en étudions aussi les limitations.

Organisation. Après une présentation de nos motivations dans le paragraphe 2, le cadre formel et la justification théorique de la méthode sont présentés dans le paragraphe 3. Le paragraphe 4 est consacré à l'implémentation de la méthode à l'aide de l'outil COSMOS. Le paragraphe 5 décrit nos expériences numériques sur deux exemples et comparent les résultats expérimentaux à ceux obtenus à l'aide d'un model

checker classique (PRISM). Le paragraphe 6 conclut en apportant des perspectives. Les preuves des différentes propriétés sont disponibles à l'adresse :
http ://www.lsv.ens-cachan.fr/∼barbot/rapport.pdf

2. Motivation et état de l'art

Le model-checking de systèmes probabilistes est à la convergence des méthodes de vérification et de l'évaluation de performances. On souhaite a priori vérifier que la probabilité de défaillance de l'alarme lors d'une panne d'un composant est suffisamment faible ou estimer le temps moyen de délivrance d'un paquet après plusieurs collusions d'un protocole de communication. Ces propriétés quantitatives sont specifiées dans des logiques qui permettent de comparer les probabilités d'un ensemble d'exécutions qui vérifient une propriété particulière avec un seuil (CSL) ou, plus généralement, de calculer l'espérance d'une variable aléatoire sur l'ensemble des exécutions éventuellement conditionnée par la satisfaction d'une formule de logique (HASL).

Une première approche, dite numérique, permet à partir d'une représentation de haut niveau du système de calculer exactement ou de façon approchée avec des algorithmes itératifs cette probabilité. Elle suppose de fortes propriétés sur le système considéré ; il doit pouvoir être modélisé formellement, le plus souvent de façon markovienne (sans mémoire) sous la forme d'une chaîne de Markov (ou plus généralement d'un processus de décision markovien)(Bianco *et al.*, 1995; Baier *et al.*, 2000), ou semi-régénératif(Amparore *et al.*, 2010). Lorsqu'elle est applicable, elle fournit un résultat fiable et précis, elle est implémentée dans des outils performants [PRISM (Kwiatkowska *et al.*, 2002), LiQuor (Ciesinski *et al.*, 2006), MRMC (Katoen *et al.*, 2009)]. Néanmoins, en raison de l'explosion combinatoire sous-jacente, sa consommation de mémoire limite significativement la taille des modèles ainsi analysables.

Pour outrepasser cette contrainte, l'approche statistique est utilisée : selon la méthode de Monte-Carlo, le modèle est simulé un grand nombre de fois dans le but de créer un échantillon d'observations suffisant pour estimer cette probabilité. Le résultat n'est plus qu'une estimation probabiliste d'un encadrement de la quantité recherchée (intervalle de confiance (Dagnelie, 2007)). Le système considéré doit aussi être doté d'une sémantique formelle en terme de processus stochastique, mais le cadre semi-régénératif n'est pas nécessaire. Vis-à-vis de la taille du modèle, l'approche numérique croît exponentiellement alors que l'approche statistique croît proportionnellement. La méthode présente aussi l'avantage d'être plus aisément parallélisable. Cette approche est aussi implémentée dans de nombreux outils [COSMOS (Ballarini *et al.*, 2011), GREATSPN (Chiola *et al.*, 1995), PRISM (Kwiatkowska *et al.*, 2002), UPPAAL (Bengtsson *et al.*, 1995), VESTA (Sen *et al.*, 2005), YMER (Younes, 2005)].

Le model-checking probabiliste s'avère crucial pour des évènements significatifs, dont les conséquences seraient désastreuses (pertes humaines, ruine financière...), mais de très faibles probabilités (par exemple, de l'ordre de 10^{-9}). De tels évènements,

dits *rares*, ne sont pas observables par simulation ; un estimateur de Monte-Carlo naïf, testant un nombre raisonnablement limité de simulations du modèle, ne peut fournir un résultat pertinent. Illustrons ce point à l'aide d'un exemple numérique. Supposons que l'on cherche à calculer une probabilité $p = 10^{-13}$ et que l'on décide de générer 10^{10} trajectoires. Avec une probabilité supérieure à 0.999 le résultat obtenu sera égal à 0 ne fournissant ainsi aucune information pertinente sur la valeur de p et avec une probabilité inférieure à 0.001, le résultat obtenu sera supérieur ou égal à 10^{-10} une estimation grossièrement erronée. Il devient donc nécessaire de recourir à des techniques dites *d'accélération* (Rubino *et al.*, 2009).

La méthode de *branchement multi-niveaux* ou *splitting* (L'Ecuyer *et al.*, 2006), consiste à inclure l'évènement rare dans une suite décroissante d'évènements intermédiaires de plus en plus rares, appelés *niveaux*, et à dupliquer ou au contraire éliminer les simulations selon que l'évènement de niveau suivant est atteint ou non, jusqu'au dernier niveau correspondant à l'évènement critique. Le passage d'un niveau à l'autre correspond à un conditionnement. La méthode d'*échantillonnage préférentiel* ou *importance sampling* (Glynn *et al.*, 1989), consiste à effectuer les simulations en modifiant la distribution, pour lequel l'évènement étudié n'est pas rare, et à pondérer les simulations obtenues de façon à corriger le biais introduit.

Ces deux techniques exigent une connaissance approfondie du système, afin de déterminer des stratégies efficaces ; le choix des niveaux pour la première, le choix de la distribution pour la seconde. Une stratégie optimale est possible mais requiert la connaissance de la solution (classiquement caractérisée par la nullité de la variance), la rendant inapplicable. Aussi pour la première technique , l'expression de la variance de l'estimateur suggère de choisir un grand nombre de niveaux. Mais cela augmente le coût de la mise en oeuvre de façon importante et nuit nettement à son efficacité. La seconde présente l'avantage d'être plus simple à mettre en oeuvre. La caractérisation de la solution optimale permet d'obtenir des heuristiques performantes pour certaines classes de problèmes : L'échantillonnage préférentiel peut être fait au niveau du modèle (*statique*) ou au niveau de la chaîne de Markov sous-jacente (*dynamique*). Afin de comparer ces différentes méthodes, on introduit des familles de systèmes à un paramètre (le nombre de clients dans une file d'attente, par exemple), et l'efficacité est mesurée en terme d'optimalité asymptotique. De Boer (de Boer, 2006) a démontré que la méthode statique ne peut construire des estimateurs asymptotiquement optimaux, même dans des cas très simples. La méthode dynamique (Srinivasan, 2002) suppose de conserver la taille de l'espace des états, ce qui retire les avantages de l'approche statistique. Pour atteindre une certaine optimalité, il est possible de calculer dans l'enveloppe convexe d'un nombre fini de distributions, une distribution optimale pour chacun des états (Dupuis *et al.*, 2007). D'autres approches empiriques s'avèrent efficaces (Heegaard *et al.*, 2007).

3. Approche générique

3.1. *Rappels*

Définition 1 *Une chaîne de Markov à temps discret (DTMC) $\mathcal{C}$, est définie par un espace d'états S, un état initial s_0, une matrice de transition $\mathbf{P}$ de dimension $S \times S$. L'état de la chaîne à l'instant n est une variable aléatoire X_n définie inductivement par $\Pr(X_0 = s_0) = 1$ et $\Pr(X_{n+1} = s' \mid X_n = s) = \mathbf{P}(s, s')$.*

En réalité, le modélisateur ne définit pas directement une chaîne $\mathcal{C}$ mais spécifie un modèle de plus haut niveau $\mathcal{M}$ (réseau de files d'attentes, réseau de Petri stochastique, etc) doté d'une sémantique formelle dont le résultat est $\mathcal{C}$.

Dans le contexte du model checking, les états de la chaîne $\mathcal{C}$ sont étiquetés par des propositions atomiques et on évalue typiquement la probabilité qu'un état s satisfasse une formule de logique temporelle aUb où U est l'opérateur *Until* et a, b sont des propositions atomiques. Afin d'évaluer cette probabilité, $\mathcal{C}$ est transformée de la façon suivante : les états qui satisfont b sont fusionnés en un état *absorbant* s_+ (i.e. $\mathbf{P}(s_+, s_+) = 1$), les états qui satisfont $\neg a \wedge \neg b$ et ceux qui satisfont $a \wedge \neg b$ mais ne permettent pas d'atteindre s_+ sont fusionnés en un état absorbant s_-. La deuxième condition est détectable par un calcul des composantes fortement connexes du graphe sous-jacent à la chaîne. Une fois cette transformation effectuée, de tout état s, la probabilité d'atteindre s_+ ou s_- est égale à 1. La probabilité recherchée est alors la probabilité d'atteindre s_+.

La méthode statistique consiste à générer K trajectoires de la chaîne qui se terminent par l'un des états absorbants. Notons K_+ le nombre de trajectoires se terminant dans l'état s_+. La variable aléatoire $\frac{K_+}{K}$ a pour espérance p et pour variance $\frac{p-p^2}{K}$ et écart-type $\sqrt{\frac{p-p^2}{K}}$. Ainsi lorsque K tend vers l'infini, cette variance tend vers 0. A l'aide des méthodes usuelles sur les variables binaires, étant données une largeur d'intervalle de confiance δ et un seuil de probabilité ε, on en déduit le nombre K de trajectoires nécessaires pour calculer un intervalle de largeur δ tel que p appartienne à cet intervalle avec une probabilité supérieure à $1 - \varepsilon$.

Nous nous plaçons ici dans le cadre où $p \ll 1$ ce qui rend la méthode inapplicable car le nombre de trajectoires à générer est trop important. Nous rappelons maintenant la méthode d'échantillonnage préférentiel afin de diminuer le nombre de trajectoires. Cette méthode consiste à appliquer lors de la génération d'une trajectoire une matrice de transition modifiée $\mathbf{P}'$ avec la restriction :

$$\mathbf{P}(s, s') > 0 \Rightarrow \mathbf{P}'(s, s') > 0 \vee s = s_- \qquad [1]$$

Autrement dit, la chaîne transformée ne peut éliminer de transitions excepté celle allant vers s_- mais peut ajouter de nouvelles transitions. La méthode maintient un facteur de correction, disons v initialisé à 1 et mis à jour lors d'une transition $s \rightarrow s'$

avec $s' \neq s_-$ de la manière suivante $v \leftarrow v\frac{\mathbf{P}(s,s')}{\mathbf{P}'(s,s')}$. Lorsque la trajectoire atteint s_-, la *valeur* de la trajectoire est 0 (comme précédemment) tandis que si la trajectoire atteint s_+ alors la valeur de la trajectoire est égale à c. Si $\mathbf{P}' = \mathbf{P}$ alors la valeur d'une trajectoire qui atteint s_+ est égale à 1. On remarque ici qu'emprunter une transition qui n'est pas présente dans $\mathcal{C}$ conduit à une valeur nulle. Par conséquent, il est plus judicieux de n'ajouter que des transitions conduisant à s_-.

On note V_s (resp. W_s) la variable aléatoire correspondant à la valeur d'une trajectoire démarrant en s dans la chaîne initiale (resp. la chaîne transformée). Par définition, $\mathbf{E}(V_{s_0}) = p$. La proposition suivante démontre la validité de la méthode.

Proposition 1 $\mathbf{E}(W_{s_0}) = p$.

Si la méthode est correcte, rien n'indique a priori comment choisir $\mathbf{P}'$ afin de diminuer la variance. La proposition suivante établit qu'il est théoriquement possible d'obtenir une variance nulle ! Notons $\mu(s)$ la probabilité d'atteindre s_+ à partir de s (et donc $\mu(s_0) = p$).

Proposition 2 *Soit* $\mathbf{P}'$ *définie par*

– $\forall s$ *tel que* $\mu(s) \neq 0$ $\mathbf{P}'(s,s') = \frac{\mu(s')}{\mu(s)}\mathbf{P}(s,s')$

– $\forall s$ *tel que* $\mu(s) = 0$ $\mathbf{P}'(s,s') = \mathbf{P}(s,s')$

Alors pour tout s*, on a* $\mathbf{V}(W_s) = 0$.

Bien évidemment, il ne s'agit que d'un résultat théorique car pour l'appliquer, il faut connaître la fonction μ, soit une information bien plus importante que la quantité à évaluer $\mu(s_0)$! L'objectif d'un échantillonnage préférentiel est donc de fournir une matrice $\mathbf{P}'$ qui diminue la variance.

3.2. *Méthode à réduction de variance garantie*

Nous décrivons d'abord les éléments théoriques nécessaires à la conception de notre méthode puis nous décrivons ses étapes.

Développements théoriques.

Définition 2 *Etant donnée une DTMC* $\mathcal{C}$*, une DTMC* $\mathcal{C}^*$ *est dite une* réduction *de* $\mathcal{C}$ *par une fonction* f *de* S*, vers* S^**, l'espace des états de* $\mathcal{C}^*$*, si en notant* $s^*_- = f(s_-)$ *et* $s^*_+ = f(s_+)$*, les assertions suivantes sont vérifiées :*

– $f^{-1}(s^*_-) = \{s_-\}$ *et* $f^{-1}(s^*_+) = \{s_+\}$.

– s^*_- *et* s^*_+ *sont des états absorbants qu'on atteint avec probabilité 1.*

– *Soit* $\mu^*(s^*)$ *pour* $s^* \in S^*$*, la probabilité d'atteindre* s^*_+ *en partant de* s^**. Alors pour tout* $s \in S$*, on a* $\mu^*(f(s)) = 0 \Rightarrow \mu(s) = 0$.

Deux états s et s' sont équivalents si $f(s) = f(s')$. Autrement dit, f^{-1} définit les classes d'équivalence de cette réduction. Nous introduisons maintenant l'hypothèse essentielle sur la distribution de probabilités μ^* afin d'obtenir un échantillonnage préférentiel à efficacité garantie dans $\mathcal{C}$.

Définition 3 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une chaîne réduite par f. $\mathcal{C}^*$ est une* réduction à variance garantie *si pour tout $s \in S$ tel que $\mu^*(f(s)) > 0$ on a :*

$$\sum_{s' \in S} \frac{\mu^*(f(s'))}{\mu^*(f(s))} \mathbf{P}(s, s') \leq 1$$

Afin de simplifier les notations, $g(s)$, pour $s \in S$, désignera la quantité $\sum_{s' \in S} \frac{\mu^*(f(s'))}{\mu^*(f(s))} \mathbf{P}(s, s')$. Nous avons maintenant tous les éléments pour obtenir un échantillonnage préférentiel efficace.

Définition 4 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une chaîne réduite par f à variance garantie. On définit $\mathbf{P}'$ une matrice de transition sur S de la façon suivante. Soit $s \in S$:*

- *Si $\mu^*(f(s)) = 0$ alors pour tout $s' \in S$, $\mathbf{P}'(s, s') = \mathbf{P}(s, s')$*
- *Si $\mu^*(f(s)) > 0$ alors pour tout $s' \in S \setminus \{s_-\}$,*
  $\mathbf{P}'(s, s') = \frac{\mu^*(f(s'))}{\mu^*(f(s))} \mathbf{P}(s, s')$ et $\mathbf{P}'(s, s_-) = 1 - g(s)$

La proposition suivante justifie le choix de $\mathbf{P}'$.

Proposition 3 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une réduction à variance garantie de $\mathcal{C}$. L'échantillonnage préférentiel de la définition 4 a les propriétés suivantes :*

– Pour tout s tel que $\mu(s) > 0$, W_s est une variable aléatoire bivaluée à valeurs dans $\{0, \mu^*(f(s))\}$.

– Par conséquent, $\mu(s) \leq \mu^*(f(s))$ et $\mathbf{V}(W_s) = \mu(s)\mu^*(f(s)) - \mu^2(s)$.

On s'intéresse à des événements rares i.e. $\mu(s) \ll 1$. Par conséquent $\mathbf{V}(W_s) \approx \mu(s)$ et dans le cas d'une réduction à variance garantie, en supposant $\mu(s) \ll \mu^*(f(s))$, on obtient $\mathbf{V}(W_s) \approx \mu(s)\mu^*(f(s))$. La variance est donc réduite d'un facteur multiplicatif $\mu^*(f(s))$. Dans le cas le plus favorable (mais peu fréquent) où $\mu(s)$ et $\mu^*(f(s))$ ont un même ordre de grandeur, la variance est encore plus réduite. Il s'agit maintenant de caractériser des situations où la réduction de variance est garantie. Dans la suite nous exhibons deux types de situation. Le premier cas repose sur la notion d'agrégation (forte).

Définition 5 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une chaîne réduite par f. $\mathcal{C}^*$ est une agrégation de $\mathcal{C}$ si :*

$$\forall s^*, s^\bullet \in S^* \; \forall s \in f^{-1}(s^*) \sum_{s' \in f^{-1}(s^\bullet)} \mathbf{P}(s, s') = \mathbf{P}(s^*, s^\bullet)$$

Le résultat suivant est l'une des nombreuses propriétés qui lient le comportement d'une chaîne et celui d'une chaîne agrégée.

Proposition 4 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une agrégation de $\mathcal{C}$. Alors $\mathcal{C}^*$ est une réduction à variance nulle.*

Notations. Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une chaîne réduite par l'identité (par conséquent $S^* = S$). Nous introduisons quelques notations utiles pour la proposition qui caractérise le deuxième cas de garantie de réduction de variance. Pour tout s,

– $Com(s)$ est l'ensemble des transitions communes à $\mathcal{C}$ et à $\mathcal{C}^*$:
$Com(s) = \{s' \mid \mathbf{P}(s,s') > 0 \wedge \mathbf{P}^*(s,s') > 0\}$.
$h(s) = \sum_{s' \in Com(s)} \mathbf{P}(s,s')$ et $h^*(s) = \sum_{s' \in Com(s)} \mathbf{P}^*(s,s')$

– $Inh(s)$ est l'ensemble des transitions de $\mathcal{C}^*$ absentes dans $\mathcal{C}$:
$Inh(s) = \{s' \mid \mathbf{P}(s,s') = 0 \wedge \mathbf{P}^*(s,s') > 0\}$.

– $Ext(s)$ est l'ensemble des transitions communes de $\mathcal{C}$ absentes dans $\mathcal{C}^*$:
$Ext(s) = \{s' \mid \mathbf{P}(s,s') > 0 \wedge \mathbf{P}^*(s,s') = 0\}$.

Proposition 5 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une chaîne réduite par l'identité. Supposons que pour tout s non absorbant tel que $\mu(s) > 0$ on ait :*

– $\forall s' \in Com(s)\ \mathbf{P}(s,s')/h(s) = \mathbf{P}(s,s')/h^*(s)$

– $\forall s' \in Inh(s)\ \mu^*(s) \leq \mu^*(s')$

– $\forall s' \in Ext(s)\ \mu^*(s) \geq \mu^*(s')$

Alors $\mathcal{C}^*$ est une réduction à variance garantie.

Principe de la méthode.

Pour calculer la probabilité $\mu(s_0)$ en utilisant un échantillonnage préférentiel, nous déterminons la distribution $\mu^\bullet$ sur une réduction à variance garantie de $\mathcal{C}$ de taille suffisamment petite pour pouvoir mener les calculs. Cette chaîne de Markov est construite en deux étapes, d'abord, en relachant des contraintes sur la chaîne initiale afin d'augmenter les symétries, puis en agrégeant les états devenus ainsi équivalents.

1) Spécifier un modèle $\mathcal{M}^*$ dont la chaîne associée $\mathcal{C}^*$ est une chaîne réduite par l'identité à variance garantie, via la proposition 5. $\mathcal{M}^*$ est un modèle intermédiaire qui ne donnera pas lieu à des calculs.

2) Spécifier un modèle $\mathcal{M}^\bullet$ dont la chaîne associée $\mathcal{C}^\bullet$ est une chaîne agrégée de $\mathcal{C}^*$. Par la proposition 4, $\mathcal{C}^\bullet$ est une chaîne réduite à variance nulle. On appellera dans la suite $\mathcal{M}^\bullet$ le modèle *simplifié* de $\mathcal{M}$.

3) Calculer numériquement la fonction $\mu^\bullet$.

4) Calculer statistiquement $\mu(s_0)$ en utilisant l'échantillonnage préférentiel de la définition 4.

Les deux dernières étapes sont automatisables. La deuxième étape l'est aussi si le formalisme spécifiant le modèle dispose d'un algorithme qui, par analyse du modèle, calcule automatiquement une chaîne agrégée. C'est le cas par exemple des réseaux de Petri bien formés (Chiola *et al.*, 1993). Seule la première étape n'est pas automatisable et requiert une analyse préalable du comportement du modèle $\mathcal{M}$ pour établir les transformations qui conduiront au modèle $\mathcal{M}^*$ (voir le paragraphe 5).

3.3. *Généralisation de la méthode*

Nous généralisons notre méthode mais cette fois-ci sans garantie de réduction de la variance.

Définition 6 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une chaîne réduite par f. On définit $\mathbf{P}'$ une matrice de transition sur S de la façon suivante. Soit $s \in S$:*

- *Si $\mu^*(f(s)) = 0$ alors pour tout $s' \in S$, $\mathbf{P}'(s, s') = \mathbf{P}(s, s')$*
- *Si $\mu^*(f(s)) > 0$ et $g(s) \leq 1$ alors pour tout $s' \in S \setminus \{s_-\}$, $\mathbf{P}'(s, s') = \frac{\mu^*(f(s'))}{\mu^*(f(s))}\mathbf{P}(s, s')$ et $\mathbf{P}'(s, s_-) = 1 - g(s)$*
- *Si $g(s) > 1$, alors pour tout $s' \in S$, $\mathbf{P}'(s, s') = \frac{\mu^*(f(s'))}{g(s)\mu^*(f(s))}\mathbf{P}(s, s')$*

Par rapport à la méthode à réduction de variance garantie, puisque l'hypothèse $g(s) \leq 1$ n'est plus imposée, nous devons « normaliser » $\mathbf{P}'(s, s')$ dans le cas défavorable. Ceci conduit à la proposition suivante sur les valeurs d'échantillonnage possibles.

Proposition 6 *Soient $\mathcal{C}$ une DTMC et $\mathcal{C}^*$ une réduction de $\mathcal{C}$. L'échantillonnage préférentiel de la définition 6 a la propriété suivante : pour tout s tel que $\mu(s) > 0$, W_s est une variable aléatoire à valeurs dans $\{0\} \uplus [\mu^*(f(s)), \infty[$.*

N'ayant pas d'information plus précise sur la distribution de W_s, la précision de l'évaluation statistique dépendra fortement de la forme de la queue de la distribution de W_s (voir le paragraphe 5.2 pour une analyse expérimentale de ce problème).

4. Implémentation

4.1. *Outils utilisés*

Pour mener nos expérimentations nous avons utilisé une version modifiée de COSMOS. COSMOS est un *Model Checker* statistique décrit dans (Ballarini *et al.*, 2011), il permet à la fois de faire du model checking et de l'évaluation de performance grâce

à une logique basée sur un automate hybride linéaire. Il utilise comme modèles des réseaux de Petri stochastiques généralisés. Sa sémantique repose sur le produit de l'automate avec le modèle, les trajectoires acceptées sont celles qui permettent d'atteindre un état final de l'automate. COSMOS permet d'estimer l'espérance de n'importe quelle variable de l'automate hybride dans un état final.

Nous avons également utilisé le model checker PRISM pour calculer numériquement la distribution de probabilités du modèle réduit et pour comparer les résultats sur les petits systèmes.

4.2. *Implémentation de la méthode*

Comme COSMOS utilise des modèles à temps continu, nous avons du adapter légèrement notre méthode. Au lieu de calculer les probabilités des transitions on calcule leurs taux. Soit un modèle $\mathcal{M}$ de sémantique $\mathcal{C}$ avec une matrice de transition P. Soit $\mathcal{M}^\bullet$ un modèle dont la chaîne associée $\mathcal{C}^\bullet$ est une chaîne agrégée d'une chaîne réduite de $\mathcal{C}$. Soit $T(x,y)$ le taux de la transition de x vers y dans $\mathcal{M}$. On a l'égalité : $P(x,y) = \frac{T(x,y)}{\sum_z T(x,z)}$. Notre méthode d'échantillonnage préférentiel est implémentée en utilisant les taux $T^\bullet$.

$$T^\bullet(x,y) = P(x,y) \cdot \frac{\mu^\bullet(y)}{\mu^\bullet(x)}$$

$$T^\bullet(x,s_-) = \begin{cases} 0 & \text{si } \sum_y P(x,y) \cdot \mu^\bullet(y) > \mu^\bullet(x) \\ 1 - \sum_y P(x,y) \cdot \frac{\mu^\bullet(y)}{\mu^\bullet(x)} & \text{sinon} \end{cases}$$

On peut remarquer que $T^\bullet$ ne définit pas une distribution de probabilité quand $\sum_y P(x,y) \cdot \mu^\bullet(y) > \mu^\bullet(x)$.

Cette adaptation nous permet d'utiliser notre méthode avec l'outil COSMOS sur des modèles *discrets*, qui sont simulés dans l'outil par des modèles continus. Des hypothèses supplémentaires sont nécessaires pour appliquer la méthode à des modèles continus dans sa généralité.

Pour calculer la valeur de chaque trajectoire nous avons modifié COSMOS pour qu'il calcule le coefficient $L(x,y) = \frac{P(x,y)}{P^\bullet(x,y)} = \frac{T(x,y)}{P^\bullet(x,y)} \cdot \frac{\sum_z T^\bullet(x,z)}{\sum_z T(x,z)}$ pour chaque transition (x,y) de $\mathcal{M}$.

L'automate de la figure 1 permet finalement de calculer la valeur des trajectoires. L'automate reste dans l'état s tant que l'état du système n'est ni s_- ni s_+, à chaque transition (x,y) la variable v est multipliée par le coefficient $L(x,y)$. Si le système atteint l'état s_- la variable v est remise à 0. Dans un état final, la variable v est donc égale à 0 si l'état s_- est atteint et à la valeur de la trajectoire si s_+ est atteint.

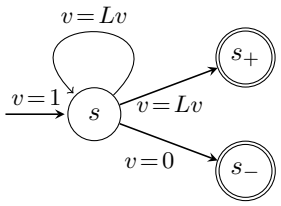


Figure 1. *Automate calculant la vraisemblance d'une trajectoire*

L'échantillonnage préférentiel augmente le coût de la simulation. Premièrement il faut calculer les probabilités $\mu^\bullet$ de $\mathcal{M}^\bullet$, ce coût est polynomial dans la taille de $\mathcal{M}^\bullet$. Ensuite à chaque transition prise par le système il faut recalculer le taux de toutes les transitions actives. Pour le calcul du taux on utilise une table de hachage pour retrouver $\mu^\bullet$. Il faut également parcourir toutes les transitions actives pour calculer $T^\bullet(x, s_-)$ dans le pire cas cela ajoute un temps de calcul linéaire dans la taille de $\mathcal{M}$ à chaque transition prise.

5. Expérimentations

Nous avons pratiqué des expérimentations sur deux modèles. Un système de ruine parallèle illustre la méthode à réduction de variance garantie. Un modèle du problème des philosophes pour lequel notre méthode ne garantit pas la réduction de la variance.

5.1. *Ruine parallèle*

Le modèle de ruine parallèle est décrit en figure 2. Seul le joueur i et ses interactions avec le joueur $i + 1$ sont représentés. N joueurs se déplacent sur une ligne de L cellules. Le joueur i dans la cellule j peut avancer en franchissant la transition $A_{i,j}$ (de taux p) si le joueur $i + 1$ n'est pas dans la cellule j ou si $i = N$. Il peut reculer en franchissant la transition $R_{i,j-1}$ de taux q. Les joueurs ne se déplacent plus lorsqu'ils sont dans la cellule 1 ou L. Ce modèle possède L^N états. On s'intéresse à la probabilité que plus de la moitié des joueurs atteignent leurs cellule L sachant qu'ils se situent initialement au milieu de leur ligne.

1. Les expérimentations ont été réalisées sur un ordinateur doté d'un processeur à 1.86Ghz et de 2Go de mémoire.

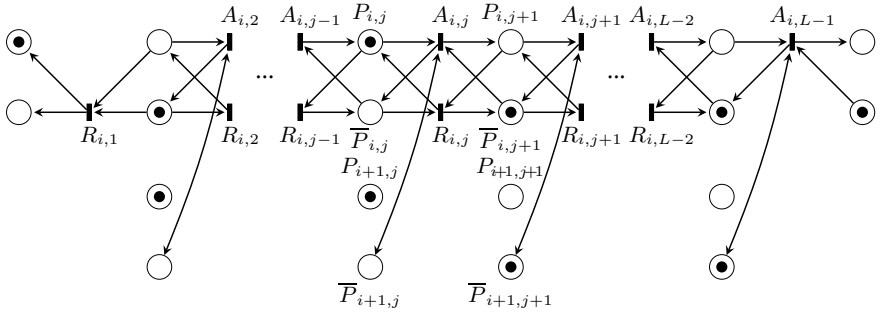


Figure 2. *Le réseau de Petri modélisant la ruine parallèle*

Ce modèle est un paradigme des problèmes de tolérance aux pannes dans lesquelles les joueurs sont les processus qui tentent d'accomplir une tache. La tache globale est accomplie lorsqu'au moins la moitié des processus ont réussi leur tâche en atteignant leur cellule 1. Les processus peuvent subir des pannes qui les éloignent de leur but allant jusqu'au blocage atteint en cellule L. Le modèle $\mathcal{M}^*$ est construit en supprimant les contraintes entre les différents joueurs. Les comportements des joueurs sont ainsi indépendants. On peut alors agréger ensemble les états qui ont le même nombre de joueurs dans chaque cellule, pour obtenir le modèle $\mathcal{M}^\bullet$ qui possède $\binom{N+L-1}{L-1}$ états. La deuxième et la cinquième colonnes du tableau 1 soulignent la réduction du nombre d'états entre $\mathcal{C}$ et $\mathcal{C}^\bullet$.

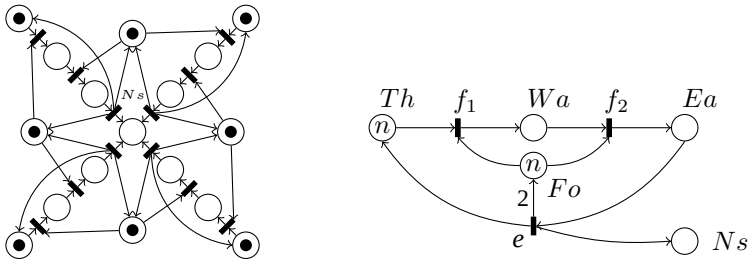
Pour ce modèle, nous avons prouvé les conditions qui garantissent la réduction de la variance (voir propositions 4 et 5). Le tableau 1 présente les résultats expérimentaux avec comme paramètres : $p = 0.3$, $q = 0.7$, $L = 15$. Nous avons simulé 300000 trajectoires quelque soit le modèle. On observe que l'estimation de $\mu(s_0)$ est toujours dans l'intervalle de confiance obtenu et que la taille de celui-ci notée δ reste toujours petite vis à vis de $\mu(s_0)$. La méthode numérique échoue par manque de mémoire dès que N est supérieur à 6. Notre méthode permet ici d'estimer une probabilité qui est à la fois hors de portée des model checkers numériques (10^{14} état pour $N = 12$) et hors de portée des méthodes statistiques sans accélération des évènements rares (**E**$(V_{s_0}) \approx 10^{-21}$ pour $N = 12$).

5.2. *Les philosophes*

Le second exemple étudié est celui du dîner des philosophes qui illustre les problèmes de concurrences pour le partage de ressources. N philosophes sont assis autour d'une table avec une fourchette entre chacun d'eux. Les philosophes ont besoin de deux fourchettes pour manger. Initialement tous les philosophes pensent. Puis un philosophe prend sa fourchette de droite (transition de taux λ_1), et ensuite sa fourchette de gauche (transition de taux λ_2). Lorsqu' il finit son repas (transition de taux ρ), il

N	taille de $\mathcal{C}$	$\mu(s_0)$	T (s) numérique	taille de $\mathcal{C}^\bullet$	$\mu^\bullet(f(s_0))$	T $\mu^\bullet$ (sec)	$\mu(s_0)$ estimé	δ	T (sec) simulation
5	7.5E5	1.884E-9	20	11628	1.444E-8	≈ 0	1.902E-09	4.142E-11	1100
6	1.1E7	1.147E-12	435	38760	2.450E-11	≈ 0	1.157E-12	3.167E-14	930
7	1.7E8	#	#	116280	5.712E-11	3	2.934E-12	8.269E-14	1200
8	2.0E9	#	#	319770	1.033E-13	14	1.885E-15	9.724E-17	1000
9	3.8E10	#	#	817190	2.323E-13	47	4.693E-15	2.776E-16	1400
10	5.7E11	#	#	1961256	4.379E-16	153	3.209E-18	3.173E-19	1000
11	8.0E12	#	#	4457400	9.626E-16	481	7.959E-18	7.664E-19	1300
12	1.3E14	#	#	9657700	1.866E-18	1000	5.590E-21	1.030E-21	1100

Tableau 1. *Résultats expérimentaux pour la ruine parallèle*



(a) $\mathcal{M}$ pour quatre philosophes

(b) le modèle réduit $\mathcal{M}^\bullet$

Figure 3. *Réseaux de Petri pour les philosophes*

se remet alors à penser. Nous nous intéressons à la probabilité que les philosophes se bloquent mutuellement en prenant tous leur fourchette de droite avant qu'il n'y ait r repas consommés. Le système est modélisé par le réseau de Petri de la figure 3(a) pour quatre philosophes. La place au centre du réseau (Ns) sert a compter le nombre de repas effectués (r).

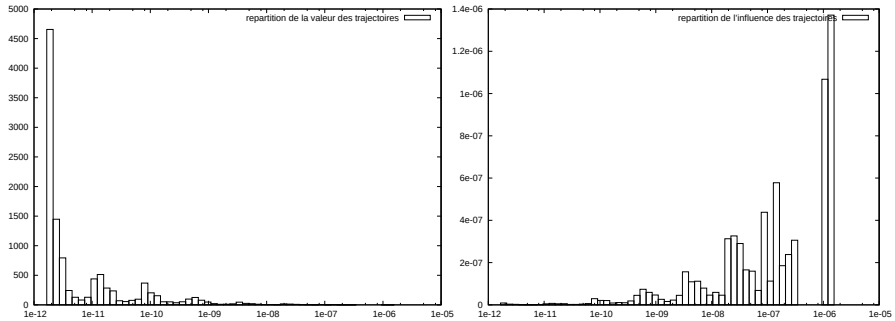
Dans le modèle réduit $\mathcal{M}^*$, les philosophes ne sont pas assis à table lorsqu'ils pensent. Lorsqu'un philosophe désire manger, il s'assied et prend successivement deux fourchettes quelconques. Le modèle $\mathcal{M}^\bullet$ agrège ensemble tous les états qui ont le même nombre de philosophes qui pensent, qui attendent et qui mangent.

Le réseau de Petri de la figure 3(b) modélise ce système avec des taux fonctionnels définis par $f_1 = m(Th) \times \lambda_1$, $f_2 = min(m(Wa), m(Fo)) \times \lambda_2$ et $e = m(Ea) \times \rho$ où m représente le marquage courant. Le nombre d'états de la chaîne $\mathcal{C}$ associé au modèle est minoré par $2^n \cdot n$ alors que celui de la chaîne $\mathcal{C}^\bullet$ est majoré par n^3. Il y a donc un saut exponentiel entre la taille de ces deux chaînes ce qui permet de résoudre numériquement la chaîne réduite alors que la chaîne originale reste hors de portée des model checkers numériques.

Les résultats expérimentaux sont présentés dans le tableau 2. Les paramètres choisis sont $r = N$, $\lambda_1 = 1$, $\lambda_2 = 100$ et $e = 10$. Pour $N = 3$ le modèle réduit est le

N	taille de $\mathcal{C}$	$\mu(s_0)$	T (s) numérique	taille de $\mathcal{C}^\bullet$	$\mu^\bullet(f(s_0))$	T $\mu^\bullet(sec)$	$\mu(s_0)$ estimé	Variance	T (s) simulation
3	56	5.822E-4	0.007	24	5.822E-4	0.009	5.822E-4	0	0.22
5	492	1.590E-6	0.028	72	9.950E-7	0.014	1.604E-6	2.651E-10	705
10	7.3E4	2.358E-11	3.45	396	1.853E-12	0.065	6.006E-12	4.614E-18	1400
15	8.8E6	3.402E-16	845	1152	3.979E-17	0.164	5.868E-19	7.466E-32	4150
18	1.4E8	#	#	1900	1.416E-19	0.327	2.212E-22	8.749E-39	6400
30	9.4E12	#	#	7936	1.566E-27	1.449	1.051E-38	1.555E-71	16000

Tableau 2. *Résultats expérimentaux pour les philosophes*



(a) Répartition de la valeur des trajectoires (b) Répartition de l'influence des trajectoires

Figure 4. *Histogramme de la valeur des trajectoires pour 10 philosophes*

même que le modèle original d'où une variance nulle. Nous avons généré un million de trajectoires quelque soit le modèle. Observons que le temps de vérification numérique est proportionnel à la taille de $\mathcal{C}$, elle-même exponentielle par rapport à N alors que le temps de vérification statistique augmente proportionnellement à N.

Il apparaît que l'inégalité de la définition 3 n'est pas vérifiée. Il existe donc des trajectoires dans le système qui possèdent une valeur supérieure à $\mu^\bullet(f(s_0))$ comme on peut l'observer sur la figure 4(a). L'existence de trajectoires avec une valeur très grande devant $\mu(s_0)$ augmente significativement la variance. Pour les grandes valeur de N on retouve ici le problème des évènements rares. Des trajectoires très rares ont un impact très grand sur l'estimation. Sur la figure 4(b) on a multiplié la valeur des trajectoires par le nombre de fois qu'elles apparaissent. On voit alors que l'unique trajectoire de valeur 2.10^{-6} a plus d'influence sur l'estimation que les 4500 trajectoires de valeur 2.10^{-12}. Les probabilités de rencontrer ces trajectoires sont très faibles ce qui conduit en général à une sous-estimation de $\mathbf{E}(V_{s_0})$ comme on l'observe dans le tableau 2 pour 10 et 15 philosophes.

6. Conclusion

Nous avons proposé une méthode de model checking statistique permettant d'évaluer un encadrement de la probabilité d'événements rares. Cette méthode se base sur la technique d'échantillonnage préférentiel. Alors que celle-ci est le plus souvent utilisée avec des heuristiques, nous proposons un cadre théorique qui permet d'en garantir la précision, sous forme d'un intervalle de confiance. A notre connaissance notre méthode est la première à garantir un tel intervalle de confiance. De plus notre méthode s'applique dans des situations très différentes. Lorsque les hypothèses associées à la méthode ne sont pas vérifiées, celle-ci fournit un résultat, mais sans garantie théorique. Nous avons implémenté les fonctionnalités nécessaires à l'outil de model checking statistique COSMOS pour intégrer notre méthode ce qui nous a permis de mener des expérimentations significatives. La garantie théorique de la diminution de la variance est un enjeu important pour la précision du résultat ; il serait intéressant de pouvoir conserver cette propriété dans un cadre plus général, en particulier sans avoir à calculer le $\mu^\bullet$ (cf. conditions de la proposition 5). Nous nous proposons d'étudier des conditions théoriques d'obtention des ces inégalités par des méthodes de couplage et d'ordre stochastique. Par ailleurs, dans le cas où la garantie de la variance n'est pas obtenue, nous voudrions établir des conditions assurant que la valeur obtenue est (avec une grande probabilité) un minorant de la valeur réelle, fait observé expérimentalement.

7. Bibliographie

Amparore E. G., Donatelli S., « Model checking CSL^{TA} with Deterministic and Stochastic Petri Nets », *DSN*, p. 605-614, 2010.

Baier C., Haverkort B. R., Hermanns H., Katoen J.-P., « On the Logical Characterisation of Performability Properties », *ICALP*, p. 780-792, 2000.

Baier C., Katoen J.-P., *Principles of Model Checking (Representation and Mind Series)*, The MIT Press, 2008.

Ballarini P., Djafri H., Duflot M., Haddad S., Pekergin N., « HASL : An Expressive Language for Statistical Verification of Stochastic Models », *(VALUETOOLS'11)*, Cachan, France, May, 2011. To appear.

Bengtsson J., Larsen K. G., Larsson F., Pettersson P., Yi W., « UPPAAL - a Tool Suite for Automatic Verification of Real-Time Systems », *Hybrid Systems*, p. 232-243, 1995.

Bianco A., Alfaro L. D., « Model Checking of Probabilistic and Nondeterministic Systems », Springer-Verlag, p. 499-513, 1995.

Chiola G., Dutheillet C., Franceschinis G., Haddad S., « Stochastic Well-Formed Colored Nets and Symmetric Modeling Applications », *IEEE Transactions on Computers*, vol. 42, n° 11, p. 1343-1360, November, 1993.

Chiola G., Franceschinis G., Gaeta R., Ribaudo M., « GreatSPN 1.7 : Graphical Editor and Analyzer for Timed and Stochastic Petri Nets », *Perform. Eval.*, vol. 24, n° 1-2, p. 47-68, 1995.

Ciesinski F., Baier C., « LiQuor : A tool for Qualitative and Quantitative Linear Time analysis of Reactive Systems. », *QEST'06*, p. 131-132, 2006.

Dagnelie P., *Statistique théorique et appliquée, volume 1*, De Boeck, 2007.

de Boer P.-T., « Analysis of state-independent importance-sampling measures for the two-node tandem queue », *ACM Trans. Model. Comput. Simul.*, vol. 16, n° 3, p. 225-250, 2006.

Dupuis P., Sezer A. D., Wang H., « Dynamic importance sampling for queueing networks », *Annals of Applied Probability*, vol. 17, p. 1306-1346, 2007.

Emerson E. A., Clarke E. M., « Characterizing Correctness Properties of Parallel Programs Using Fixpoints », *ICALP*, LNCS 85, 1980.

Glynn P. W., Iglehart D. L., « Importance sampling for stochastic simulations », *Management Science*, 1989.

Heegaard P. E., Sandmann W., « Ant-based approach for determining the change of measure in importance sampling », *Winter Simulation Conference*, p. 412-420, 2007.

Katoen J.-P., Zapreev I. S., Hahn E. M., Hermanns H., Jansen D. N., « The Ins and Outs of the Probabilistic Model Checker MRMC », *Quantitative Evaluation of Systems, International Conference on*, vol. 0, p. 167-176, 2009.

Kwiatkowska M., Norman G., Parker D., « PRISM : Probabilistic Symbolic Model Checker », *in* T. Field, P. Harrison, J. Bradley, U. Harder (eds), *Computer Performance Evaluation : Modelling Techniques and Tools*, vol. 2324 of *LNCS*, Springer Berlin / Heidelberg, p. 113-140, 2002.

Kwiatkowska M., Norman G., Parker D., « Stochastic Model Checking », *in* M. Bernardo, J. Hillston (eds), *SFM'07*, vol. 4486 of *LNCS)*, Springer, p. 220-270, 2007.

L'Ecuyer P., Demers V., Tuffin B., « Splitting for rare-event simulation », *Winter Simulation Conference*, p. 137-148, 2006.

Legay A., Delahaye B., Bensalem S., « Statistical model checking : an overview », *Proceedings of the First international conference on Runtime verification*, RV'10, Springer-Verlag, p. 122-135, 2010.

Rubino G., Tuffin B., *Rare Event Simulation using Monte Carlo Methods*, Wiley, 2009.

Sen K., Viswanathan M., Agha G., « VESTA : A Statistical Model-checker and Analyzer for Probabilistic Systems », *QEST*, vol. 0, p. 251-252, 2005.

Srinivasan R., *Importance sampling – Applications in communications and detection*, Sringer Verlag, Berlin, 2002.

Younes H., « Ymer : A Statistical Model Checker », *in* K. Etessami, S. Rajamani (eds), *Computer Aided Verification*, vol. 3576 of *LNCS*, Springer Berlin / Heidelberg, p. 171-179, 2005.