Vérification Probabiliste et Approximation

Richard Lassaigne
Logique mathématique,
CNRS-Université Paris 7

ANR Sécurité et Sûreté Informatique
Projet VERAP
http ://www.lri.fr/∼mdr/verap

Algorithmique et Complexité (LRI, Orsay)

Frédéric Magniez, Michel de Rougemont, Mathieu Tracol

Logique mathématique (Paris 7)

Richard Lassaigne

Parallélisme, Grand Large (LRI, INRIA Orsay)

Sylvain Peyronnet, Thomas Hérault

Visiteur

Eldar Fisher (Technion, Israël)

Contexte : Approximation de la vérification

Test de propriétés et d'équivalence

Complexité de la vérification probabiliste

Schémas probabilistes d'approximation

Conclusion

Approximation de la vérification

Motivation :

Complexité de la vérification par model checking

Approximation sur les données/Approximation sur le calcul

Objectifs :

- Approximation de la satisfaction d'une propriété logique
- Algorithmes d'approximation efficaces et robustes

Méthodes :

- Property Testing (Approximation sur les données)
- Schémas probabilistes d'approximation
  (Approximation sur le calcul)

Approximation **efficace** de la satisfaction et de l'équivalence logiques :

$$\text{Model } \mathcal{M} \models_\varepsilon \text{ Property}$$

Approximation sur les données :

Modèle = **Automate** $\mathcal{A}$

Propriété = un mot $w \in_\varepsilon \mathcal{L}(\mathcal{A})$

Test de propriétés

(E. Fisher, F. Magniez and M. de Rougemont)

Approximation sur le calcul :

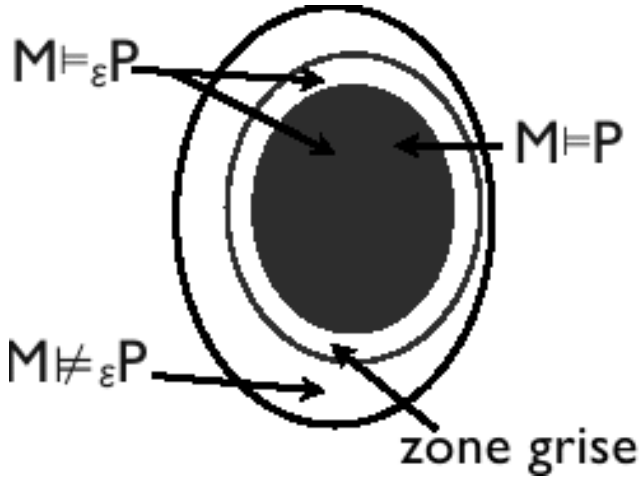
Modèle = **Système de transition probabiliste**

Propriété = la **mesure de probabilité** de ... **égale**$_\varepsilon$ **p**

Schémas probabilistes d'approximation

(R. Lassaigne, S. Peyronnet)

- Satisfaction classique : $\mathcal{M} \models \mathbf{P}$
- Décider si $\mathcal{M}$ satisfait la propriété $\mathbf{P}$



- Satisfaction approchée : $\mathcal{M} \models_\varepsilon \mathbf{P}$

  $\mathcal{M} \models_\varepsilon \mathbf{P}$ si $\mathcal{M}$ est ε-proche de $\mathcal{M}'$ tel que $\mathcal{M}' \models \mathbf{P}$

Testeur : Algorithme probabiliste A

- si $\mathcal{M} \models \mathbf{P}$, alors A accepte
- si $\mathcal{M}$ est ε-loin de $\mathbf{P}$, alors A rejette avec grande probabilité

  Le temps de calcul est indépendant de $|\mathcal{M}|$

  (mais dépend de $1/\varepsilon$)

Distance d'édition avec déplacement

- Distance d'édition avec déplacement : $dist(w, w')$
  - Insertion, suppression, modification
  - Déplacement de blocs

  $w = 01110001111011001$

  $w' = 01110111100011000$ $\qquad dist(w, w') = 2$

  Si L est un langage, $d(w, L) = min_{w' \in L}\{d(w, w')\}$

- Calcul exact : NP-difficille
- Calcul approché : $log|w|$-approximation, temps quasi-linéaire

- ε-test de l'égalité :
  - accepte si $w = w'$
  - rejette si la distance normalisée $\frac{dist(w,w')}{max(|w|,|w'|)} > \varepsilon$

Statistiques uniformes d'un mot

Mots de longueur n, $n = 12$

$n - k + 1$ blocs de longueur $k = 1/\varepsilon$ $w = 001010101110$

$$u.stat(w) = 1/(n-k+1) \cdot \begin{pmatrix} \sharp n_1 \\ \dots \\ \dots \\ \sharp n_{2^k} \end{pmatrix} \quad u.stat(w) = 1/11 \cdot \begin{pmatrix} 1 \\ 4 \\ 4 \\ 2 \end{pmatrix}$$

n_1 nombre de $00\dots0$ Pour $k = 2$, $n - k + 1 = 11$

n_2 nombre de $00\dots1$

$\dots$

n_{2^k} nombre de $11\dots1$

La distance normalisée $\frac{dist(w,w')}{n} \approx |u.stat(w) - u.stat(w')|$
lorsque les mots sont de longueur proche

Approximation de la distance par les statistiques

Propriétés :

- Correction

  Si deux mots w, w' sont proches,

  alors les statistiques $u.stat(w)$ et $u.stat(w')$ sont proches

$$dist(w, w') \leq \varepsilon^2.n \Rightarrow |u.stat(w) - u.stat(w')| \leq 6.1\varepsilon$$

- Robustesse

  Si deux mots w, w' sont éloignés,

  alors les statistiques $u.stat(w)$ et $u.stat(w')$ sont éloignées

$$dist(w, w') \geq 5\varepsilon.n \Rightarrow |u.stat(w) - u.stat(w')| \geq 6.5\varepsilon$$

Méthode :

- Echantillonner N sous-mots de longueur $k = 1/\varepsilon$,
- Estimer les statistiques uniformes de w, w' par $Y(w)$ et $Y(w')$

$$Y(w) = 1/N \sum_{i=1}^{N} X_i \quad Y(w') = 1/N \sum_{i=1}^{N} X'_i \quad X_i = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \dots \\ 0 \end{pmatrix}$$

Lemme(Chernoff-Hoeffding) : $Y(w)$ approximates $u.stat(w)$

Corollaire : $Y(w) - Y(w')$ approximates $dist(w, w')/n$

ε-Testeur de l'égalité :

Si $|Y(w) - Y(w')| < \varepsilon$, alors accepter, sinon refuser

Appartenance à un langage régulier

A automate à m états sur Σ : A^k automate à m états sur Σ^k

Lemme : Si $w \in L$, alors w est ε-proche de $u.v_1 \dots v_l$ où $|u|, |v_i| \leq m$ et $\{v_1, \dots, v_l\}$ est un ensemble de boucles A^k-compatibles

Pré-calcul :

- Calculer les statistiques des boucles A^k-compatibles
- Considérer la réunion H des enveloppes convexes des statistiques associées aux $\{v_1, \dots, v_t\}$ pour $t = |\Sigma|^k + 1$
- Soit H_ε l'ensemble des points de la grille de pas $\frac{\varepsilon}{|\Sigma|^k}$ à distance $\leq \varepsilon/2$ de H

Proposition : Le temps de calcul de H_ε est $O(m^{|\Sigma|^k})$

ε-Testeur de l'appartenance :

Calculer l'approximation $Y(w)$ de la statistique uniforme du mot

Si $dist_1(Y(w), H_\varepsilon) < \varepsilon$, alors accepter, sinon refuser

Gain en Complexité

- Reconnaissance d'une entrée
  Temps linéaire $\longrightarrow$ Temps indépendant de la longueur du mot

- Appartenance à un langage régulier
  Temps exponentiel $\longrightarrow$ Temps polynomial

- Equivalence de 2 automates
  PSPACE-complet $\longrightarrow$ Temps polynomial

- Equivalence de 2 automates à pile
  Indécidable $\longrightarrow$ Temps exponentiel

Résultats : Fisher, Magniez et de Rougemont [LICS06]

- Approximation de la satisfaction et de l'équivalence logiques
- Testeur pour la distance sur les mots
- Testeur pour l'appartenance à un langage régulier
- Extensions aux langages context-free et aux langages d'arbres
- Algorithmes probabilistes simples, mais preuves difficilles

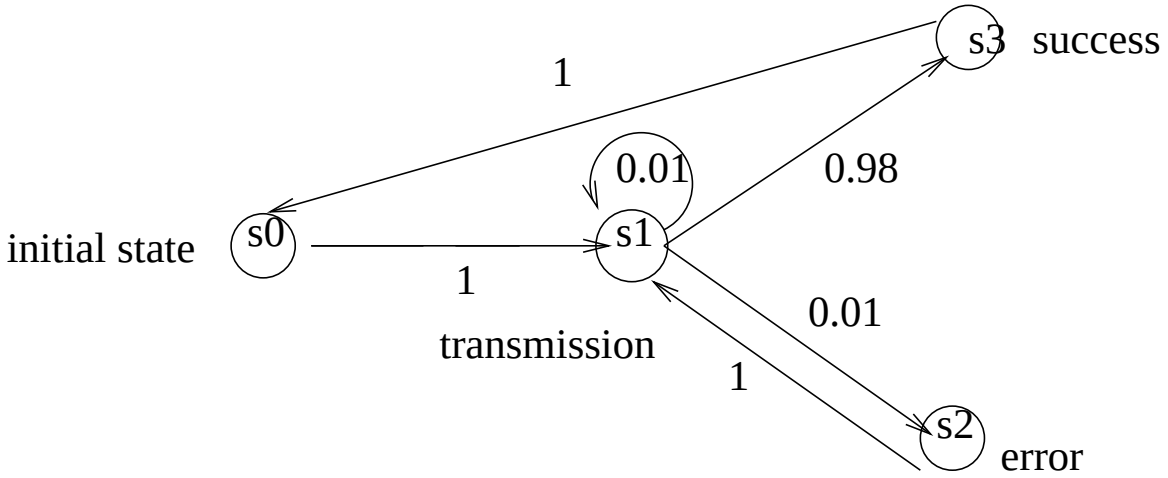
Applications : [ICDT07] [WAIM08]

- Notion de testeur tolérant $\Longrightarrow$ Possibilité de correction
- Testeur pour la distance entre deux fichiers XML
- Correcteur pour un fichier proche d'une DTD
- Validité approchée pour le streaming de documents XML

Systèmes de Transition Probabilistes

Entrée :

- Modèle $\mathcal{M} = (S, P, L)$ et état initial s_0
- $P : S^2 \longrightarrow [0, 1]$ Fonction de probabilité
- $L : S \longrightarrow 2^{AP}$ (étiquetage des états)
- Formula ψ (**LTL**)



Sortie : $Prob_{\Omega}[\psi]$

Exemple : $\psi \equiv transmission$ Until $success$

(Ω espace probabiliste des chemins d'exécution d'origine s_0)

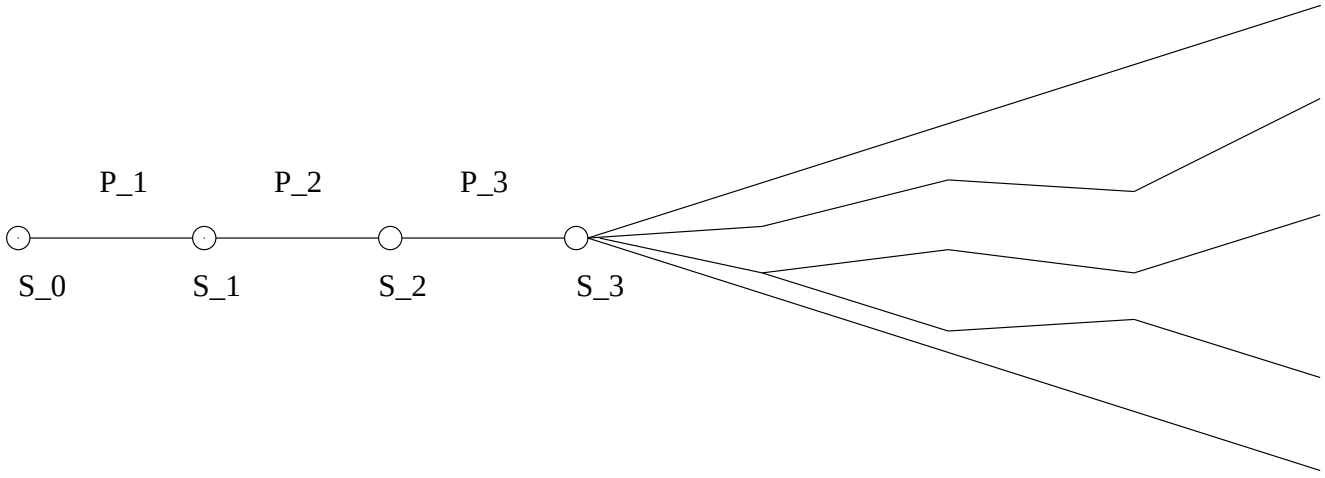
Espace probabiliste :

Cône des extensions d'un chemin fini $\rho = (s_0, s_1, \ldots, s_n)$:

$$Prob(\{\sigma/\sigma \text{ est un chemin et } (s_0, s_1, \ldots, s_n) \text{ est un préfixe de } \sigma\}) =$$

$$\prod_{i=1}^{n} P(s_{i-1}, s_i)$$

La mesure peut être définie sur la famille borélienne engendrée par les ensembles $\{\sigma/\rho$ préfixe de $\sigma\}$ où ρ est un chemin fini.



L'ensemble des chemins $\{\sigma/\sigma(0) = s_0$ et $\mathcal{M}, \sigma \models \psi\}$ est mesurable.

Complexité : Courcoubetis et Yannakakis [CY95]

Vérification qualitative (i.e. $prob > 0$?)

Même **complexité** que **model checking pour LTL**

$$O(|M|.2^{|\psi|})$$

Vérification quantitative (i.e. $prob =?$)

$$O(poly(|M|).2^{|\psi|})$$

Méthode : Calculer $Prob_{\Omega}[\psi]$

- Transformer étape par étape la formule
  et la chaîne de Markov $\mathcal{M}$
- Eliminer les connecteurs temporels un par un
- En préservant la probabilité de satisfaction
- En résolvant un système d'équations linéaires de taille $|M|$.

Problème :

- Phénomène d'explosion combinatoire (dû à la modélisation)
- Le problème n'est pas le temps, mais l'espace utilisé pour la représentation explicite du modèle

Exemple : PRISM (Probabilistic Model Checker)

Protocole (probabiliste) du diner des philosophes

Méthodes classiques :

- Représentation symbolique (OBDD)
- Méthodes basées sur l'utilisation de SAT-solvers (Bounded Model Checking)
- Abstraction

Notre solution :

- Méthode d'approximation
- Représentation succincte du modèle

Problèmes d'**énumération** : L. Valiant [Val79]

- $\sharp P$ classe des problèmes d'énumération associés aux problèmes de décision NP
- $\sharp SAT$ est un problème $\sharp P$-complet

Schéma probabiliste d'approximation :

(R. Karp et M. Luby 85)

Algorithme probabiliste A

- Entrée : instance x d'un problème d'énumération, $\varepsilon, \delta > 0$
- Sortie : valeur $A(x, \varepsilon, \delta)$ telle que

$$Pr[(1-\varepsilon)\#(x) \leq A(x,\varepsilon,\delta) \leq (1+\varepsilon)\#(x)] \geq 1-\delta$$

Schéma probabiliste d'approximation pleinement polynomial : **FPRAS**

Le temps de calcul est $poly(|x|, (1/\varepsilon), \log(1/\delta))$

Schémas probabilistes d'approximation classiques :

- Approximation de $\sharp DNF$ Karp, Luby, et Madras [KLM89]

Entrée : Formule propositionnelle sous forme normale disjonctive Φ
Sortie : Nombre de valuations satisfaisant Φ

- Approximation de la fiabilité d'un réseau D. Karger [K99]

Entrée : Graphe dont les arêtes ont une probabilité de disparition
Sortie : La probabilité que le graphe reste connexe

Peut-on approximer efficacement $Prob_\Omega(\psi)$?

Cas général : R. Lassaigne et S. Peyronnet [APAL08]

L'existence d'un schéma probabiliste d'approximation pleinement polynomial pour calculer $Prob_\Omega(\psi)$ ($\psi \in LTL$)
entrainerait que $RP = NP$.

RP **:** Classe de complexité des problèmes décidables par un algorithme probabiliste de Monte-Carlo (erreur d'1 seul côté).

Randomized **P**olynomial time : languages L t.q.
il existe un algorithme probabiliste A en temps polynomial

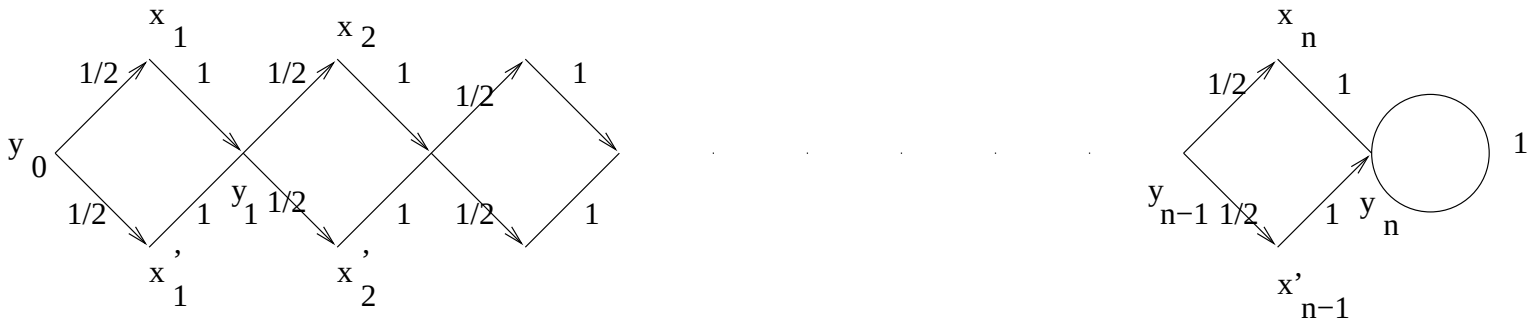
$$x \in L \ : \ Prob[\text{ A accepte } x] \geq 1/2$$
$$x \notin L \ : \ Prob[\text{ A accepte } x] = 0$$

$\#SAT$ peut se réduire à l'énumération des chemins de longueur $2n$, dont les extensions infinies satisfont une certaine formule LTL.

Instance de $\#SAT$:

- variables propositionnelles $\{x_i \ / \ 1 \leq i \leq n\}$
- clauses propositionnelles : $c_1, \ldots, c_m$

Formule LTL ψ : $\bigwedge_{i=1}^{n} F c_j$



Etiquettes des états :

$L(x_i) = \{c_j \ / \ x_i$ apparaît dans $c_j\}$ $(i = 1, \ldots, n)$

$L(x'_i) = \{c_j \ / \ \neg x_i$ apparaît dans $c_j\}$ $(i = 1, \ldots, n)$

Idée de la preuve :

- Compter le nombre de ces chemins fournit $Prob_{\Omega}(\psi)$
- S'il existait un FPRAS pour calculer $Prob_{\Omega}(\psi)$, alors il existerait un algorithme probabiliste d'approximation pour $\#SAT$ en temps polynomial
- Un FPRAS pour $\#SAT$ permettrait de distinguer, en temps polynomial, pour l'entrée x, entre $\#(x) = 0$ et $\#(x) > 0$
- Alors il existerait un algorithme probabiliste en temps polynomial pour SAT et $NP \subseteq BPP$

BPP classe des algorithmes Monte-Carlo avec erreur des 2 côtés

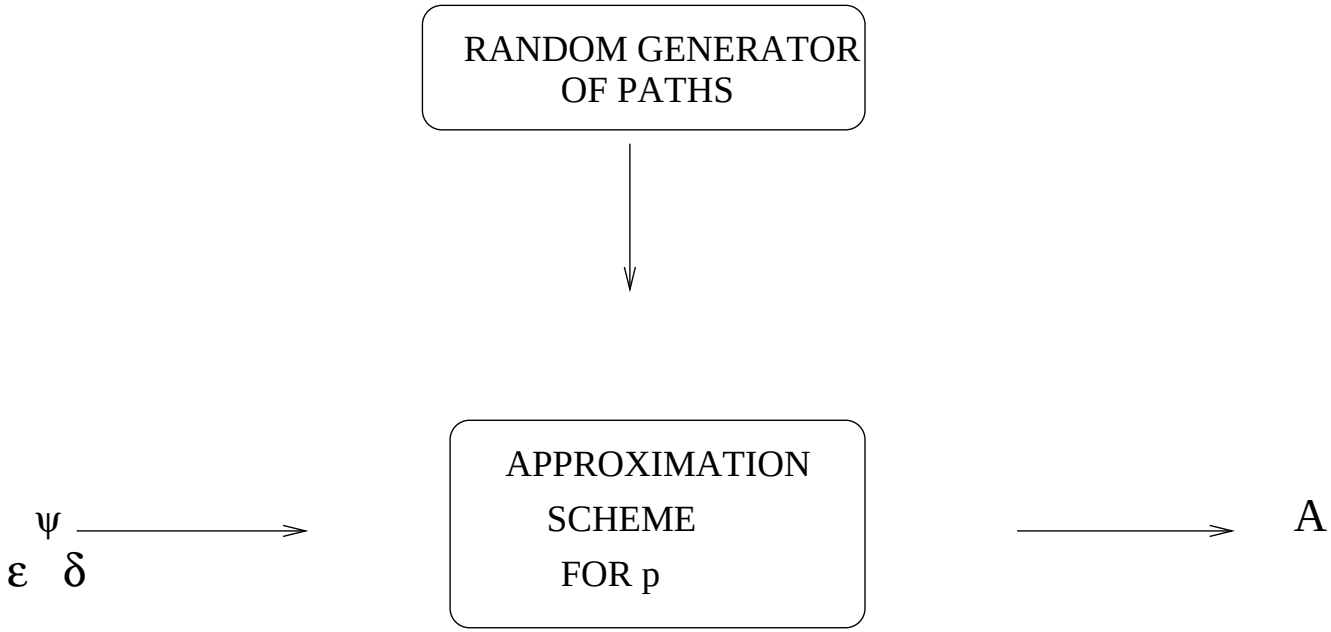
D'autre part :

$$NP \subseteq BPP \Rightarrow RP = NP$$

Idée de Jerrum et Sinclair

Schéma probabiliste d'approximation

On désire approximer une probabilité p.



$$Pr[(p-\varepsilon) \leq A \leq (p+\varepsilon)] \geq 1-\delta$$

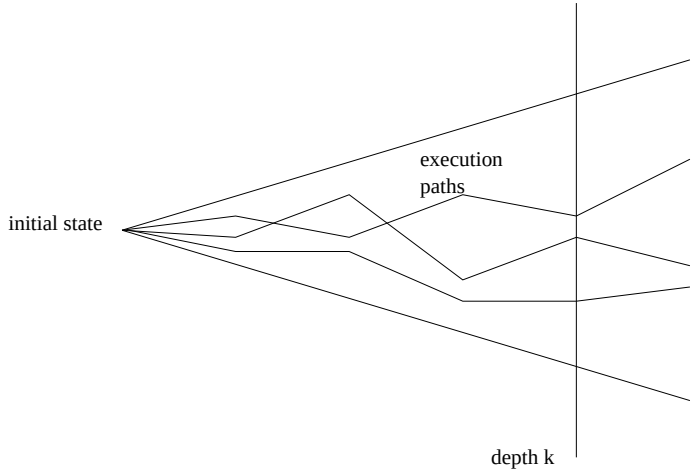
ε : paramètre d'approximation (additive)

δ : paramètre de confiance (algorithme probabiliste)

Le schéma est dit pleinement polynomial si le temps est $poly(|\text{entrée}|, (1/\varepsilon), \log(1/\delta))$

On considère $Prob_k(\psi)$ où :

- l'espace probabiliste est l'espace des chemins de longueur $\leq k$



- ψ exprime une propriété monotone (accessibilité)

$$\lim_{k \to \infty} Prob_k(\psi) = Prob_\Omega(\psi)$$

- si ψ exprime une propriété anti-monotone (sûreté), on considère $\neg\psi$

Algorithme générique d'approximation $\mathcal{GAA}$

input : $\phi, diagramme, \varepsilon, \delta$

$A := 0$

$N := \log(\frac{2}{\delta})/2\varepsilon^2$

Pour $i := 1$ à N

- Engendrer de manière aléatoire un chemin σ de longueur k
- Si ψ est vraie sur σ alors $A := A + 1$

Retourner (A/N)

Algorithme basé sur une estimation de type Monte-Carlo et la borne de Chernoff-Hoeffding

Diagramme : représentation succincte du système
(par exemple programme dans le langage d'entrée d'APMC ou de PRISM)

Méthode :

Estimation (Monte-Carlo) + borne de Chernoff-Hoeffding

X variable de Bernoulli $(0, 1)$ avec probabilité de succès p

- Faire N tirages aléatoires independants $X_1, X_2, \ldots, X_N$
- Estimer p par $\mu = \sum_{i=1}^{N} X_i/N$ avec erreur ε
- La taille de l'échantillon N est telle que la probabilité d'erreur (de l'algorithme) $< \delta$

Borne de Chernoff-Hoeffding :

$$Pr[\mu < p - \varepsilon] + Pr[\mu > p + \varepsilon] < 2e^{-2N\varepsilon^2}$$

Si $N \geq \ln(\frac{2}{\delta})/2\varepsilon^2$, alors

$$Pr[p - \varepsilon \leq \mu \leq p + \varepsilon)] \geq 1 - \delta$$

Théorème :

$\mathcal{GAA}$ est un FPRAS pour $Prob_k(\psi)$

Méthodologie : Pour approximer $Prob_\Omega(\psi)$

- Choisir $k \approx log|M| \cdot ln(1/\varepsilon)$
- Itérer l'approximation de $Prob_k(\psi)$

Corollaire :

L'algorithme de point fixe obtenu en itérant $\mathcal{GAA}$ est un schéma probabiliste d'approximation en espace logarithmique pour $Prob(\psi)$

Remarque :

- La complexité en espace est logarithmique...
- La vitesse de convergence est régie par l'ordre de multiplicité de la 2e valeur propre (thm de Perron-Frobenius)

Résultats : [WoLLIC05] [QEST06] [APAL08]

- Approximation quantitative de l'accessibilité et de la sûreté
- L'existence d'un algorithme général d'approximation en temps polynomial entraînerait $RP = NP$
- Réduction exponentielle de la complexité en espace
- Extension aux chaînes de Markov en temps continu
- Un outil de vérification approchée efficace (APMC)
- Implémentation distribuée
- Nombreux cas d'étude (algorithmes distribués, protocoles de communication, réseaux de capteurs, modèles biologiques,...)
- Intégration avec le model checker probabiliste **PRISM** (Oxford)

- Théorie de l'approximation pour la vérification
- Correction et robustesse de l'approximation
  sur les données pour la vérification classique

- Efficacité théorique des schémas d'approximation
  pour la vérification probabiliste
- Efficacité pratique de l'outil APMC

Perspectives

- Approximation pour les Processus de Décision Markoviens
- Métriques pour l'équivalence de trace

- [CY95] C. Courcoubetis and M. Yannakakis. *The complexity of probabilistic verification*. JACM, 24(4), p. 857-907, 1995.

- [LICS06] E. Fisher, F. Magniez and M. de Rougemont. *Approximate Satisfiability and Equivalence*. Proc. 21st IEEE Symposium on Logic In Computer Science, 2006.

- [VMCAI04] T. Hérault, R. Lassaigne, F. Magniette and S. Peyronnet. *Approximate Probabilistic Model Checking*. Proc. 5th Int. Conf. on Verification, Model Checking and Abstraction, LNCS n° 2937, p. 73-84, 2004.

- [QEST06] T. Hérault, R. Lassaigne, and S. Peyronnet. *APMC 3.0 : Approximate verification of Discrete and Continuous Time markov Chains*. Proc. 3rd Int. Conf. on Quantitative Evaluation of Systems, 2006.

- [K99] D.R. Karger. *A Randomized Fully Polynomial Approximation Scheme for the All Terminal Network Reliability Problem*. SIAM Journal on Computing 29(2), p.492-514, 1999.

- [KLM89] R. Karp, M. Luby and N. Madras. *Monte-Carlo Approximation Algorithms for Enumeration Problems.* Journal of Algorithms 10, 429-448, 1989.

- [APAL08] R. Lassaigne et S. Peyronnet. *Probabilistic Verification and Approximation*. Journal of Pure and Applied Logic, vol. 152 (1-3), p. 122-131, 2008.

- [LR04] R. Lassaigne et M. de Rougemont : *Logic and Complexity.* Springer-Verlag, 360 p., 2004.

- [Val79] L. Valiant. *The complexity of enumeration and reliability problems*. SIAM Journal of Computing 8, p.410-421, 1979.

- [Var85] *Automatic verification of probabilistic concurrent finite-state programs* Proc. 26th IEEE FOCS, p. 327-338, 1985.