

REVUE DE STATISTIQUE APPLIQUÉE

ALI HAMLILI

Estimation non-paramétrique par régression antitone de la croissance de fiabilité des logiciels

Revue de statistique appliquée, tome 48, n° 3 (2000), p. 39-56

http://www.numdam.org/item?id=RSA_2000__48_3_39_0

© Société française de statistique, 2000, tous droits réservés.

L'accès aux archives de la revue « *Revue de statistique appliquée* » (<http://www.sfds.asso.fr/publicat/rsa.htm>) implique l'accord avec les conditions générales d'utilisation (<http://www.numdam.org/conditions>). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

NUMDAM

Article numérisé dans le cadre du programme
Numérisation de documents anciens mathématiques

<http://www.numdam.org/>

ESTIMATION NON-PARAMÉTRIQUE PAR RÉGRESSION ANTITONE DE LA CROISSANCE DE FIABILITÉ DES LOGICIELS

Ali Hamlili

ENSIAS, LMD, B.P. 713, Agdal, Rabat, Maroc

e-mail : hamlili@ensias.ac.ma

RÉSUMÉ

Dans cet article, nous nous intéressons à l'étude statistique de la fiabilité des logiciels au cours des phases de test et de maintenance. Ainsi, nous proposons une méthode dynamique d'estimation et de prédiction de la croissance de la fiabilité des logiciels qui se base sur l'outil statistique de la régression antitone. Cette méthode nous permet d'étudier l'évolution de la croissance de fiabilité sans introduire d'hypothèses trop restrictives. Pour ce faire, nous montrons que l'estimateur du maximum de vraisemblance sous des restrictions d'ordre est équivalent à l'estimateur obtenu par la régression antitone. Ce qui nous amène à définir la notion d'information de Kullback-Leiber cumulée sur le cône convexe des fonctions antitones pour la famille des processus exponentiels. Enfin, nous confrontons notre modèle à cinq modèles paramétriques classiques, les plus cités dans la littérature, sur des jeux de données émanant de projets réels de fiabilité des logiciels et nous comparons les capacités prédictives des modèles en utilisant la méthode du u-plot.

Mots-clés : *Fiabilité des Logiciels, modèle de croissance de fiabilité, intensité conditionnelle de défaillance, régression antitone, information de Kullback-Leiber, u-plot.*

ABSTRACT

In this paper, we consider the statistical evaluation of the software reliability during phases of test and maintenance. Thus, we propose here a dynamic method for estimating and predicting the software reliability growth based on the statistical tool of the antitone regression. This method is going to allow us to study the evolution of the reliability growth without introducing too restrictive hypotheses. To make this, we show that the maximum likelihood estimate under order restrictions is equivalent to the estimate obtained by the antitone regression. That will bring us to define the notion of cumulated Kullback-Leiber information on the convex cone of antitone functions for the exponential families. Finally, we confront our model with five classic parametric models, the most quoted in the literature, on data sets emanating from real software reliability projects and we compare their predictive and replicative capacities by using the u-plot method.

Keywords : *Software reliability, reliability growth model, conditional failure intensity, antitonic regression, Kullback-Leiber information, u-plot.*

1. Introduction et hypothèses préliminaires

L'aptitude à produire de bons logiciels est à présent considérée comme un atout incontestable à tous les niveaux de la technologie moderne. En effet, avec l'utilisation intensive de l'outil informatique, la mesure de la qualité du logiciel et la sûreté de fonctionnement des systèmes informatiques sont devenues des sujets de préoccupation autant pour le développeur que pour l'utilisateur. La norme ISO 9000-3 décrivant les lignes directrices pour l'application de l'ISO 9001 à l'installation, la mise à disposition et la maintenance des logiciels préconise l'application des techniques statistiques et l'introduction de la fiabilité pour caractériser ce type de produits [ISO-98]. L'étude statistique de la fiabilité des logiciels a débuté dès les années soixante-dix avec les travaux pionniers de Moranda [MOR-72] [MOR-75], Jelinski [JEL-72], Shooman [SHO-72], Coutinho [COU-73], Littlewood et Verrall [LIT-73], Schick et Wolverton [SCH-78] etc.

1.1. Modèle de test d'un logiciel et détection des fautes

Dans le cadre du test fonctionnel des logiciels, le programme subit une série d'exécutions en vue de détecter d'éventuelles fautes de conception et de les éliminer. Dans ce contexte, une défaillance se définit comme étant un état du logiciel dans lequel, suite à une entrée, celui-ci n'est plus en mesure de rendre des résultats conformes à ceux prévus lors de sa conception. On ne considère ici que des défaillances dues à des fautes de conception.

Hypothèse 1 – *La détection des fautes dans l'approche du test fonctionnel et de la maintenance corrective des logiciels se fait à travers la manifestation aléatoire des défaillances au cours de l'exécution du programme.*

1.2. Processus des défaillances-corrrections

Dans le cadre de l'étude statistique de la fiabilité des logiciels, le processus des défaillances-corrrections induit par le test fonctionnel ou la maintenance corrective peut être décrit dynamiquement [SOL-96] à l'aide :

- du processus des défaillances : qu'on peut définir indifféremment par :
 - le processus $\{T_i/i \in \mathbb{N}^*\}$ des instants successifs de défaillance tels que $T_1 < T_2 < \dots < T_i < \dots$,
 - le processus $\{X_i/i \in \mathbb{N}^*\}$ des temps interdéfaillances tels que $X_i = T_i - T_{i-1}$ (avec $T_0 = 0$),
 - le processus $\{N_t/t \in \mathbb{R}^+\}$ des nombres de défaillances survenues avant l'instant t . Il peut être défini à partir du processus des instants des défaillances du logiciel $N_t = \sum_{i=1}^{\infty} \mathbb{1}_{[0,t]}(T_i)$.
- du processus des corrections que l'on peut définir, de manière similaire, indifféremment par le processus des instants de correction, le processus des temps

intercorrections ou encore par le processus des nombres de corrections effectuées avant l'instant t .

Hypothèse 2 – *Les corrections sont opérées ponctuellement et immédiatement après l'observation d'une défaillance.*

Cette hypothèse signifie que les processus des défaillances et des corrections sont ici confondus. La théorie des processus ponctuels [SNY-75] permet de caractériser la loi du processus à partir de l'intensité conditionnelle de défaillance.

1.3. Croissance de fiabilité et intensité de défaillance

Le standard IEEE définit la fiabilité des logiciels comme étant la probabilité que le logiciel ne cause pas la défaillance d'un système pendant un temps déterminé et sous des conditions spécifiées [IEEE94]. Formellement, la fiabilité peut être exprimée à tout instant t par

$$\forall t > 0, \forall x \geq 0 : R_t(x) = P(T_{N_{t+1}} - t > x) = \exp\left(-\int_t^{t+x} \lambda_s ds\right) \quad (1)$$

avec $\lambda_t = \lim_{h \rightarrow 0} \frac{1}{h} P(N_{t+h} - N_t = 1 / \mathcal{H}_t)$ où \mathcal{H}_t est la tribu des événements survenus avant l'instant t . Le paramètre λ_t est appelé «intensité de défaillance conditionnelle à l'instant t ».

Construire un modèle statistique de fiabilité des logiciels consiste essentiellement à se donner une famille de lois de probabilité plausibles pour le processus stochastique caractérisant le phénomène des défaillances-corrrections. Pour les modèles de prédiction de la fiabilité des logiciels, cela revient, à donner une expression ou une forme à l'intensité conditionnelle de défaillance. Aussi, dans la modélisation du processus de correction, on suppose la croissance de fiabilité du logiciel [COU-73] [LIT-73] [YAM-83] [SOL-96].

Hypothèse 3 – *Le processus des corrections entraîne une croissance de fiabilité du logiciel.*

Dans la littérature, on retrouve plusieurs modélisations de la croissance de fiabilité qui ont été proposées, discutées, modifiées et ensuite généralisées, alors que d'autres ont été critiquées et rejetées [LIT-73] [SCH-78] [YAM-83] [SHO-84] [MUS-87] [XIE-91] [LYU-96].

1.4. Durée moyenne instantanée de bon fonctionnement

Par ailleurs, notons que dans les approches classiques, on définit la durée moyenne instantanée de bon fonctionnement du logiciel ou MTTF (Mean Time To Failure), en tant qu'indicateur sur le temps qu'il faut attendre pour observer une

défaillance

$$MTTF_t = E(T_{N_{t+1}} - t) = \int_0^{+\infty} R_t(x) dx \quad (2)$$

qui, si nous nous plaçons à l'instant de la $n^{\text{ème}}$ défaillance, s'écrit

$$MTTF_{T_n} = E(X_{n+1}) \quad (3)$$

1.5. Modélisation de la croissance de fiabilité des logiciels

Comme le logiciel ne subit pas de modifications entre deux instants successifs de correction et qu'il ne s'use pas, il est naturel de considérer que l'intensité de défaillance reste constante pendant cet intervalle de temps.

Hypothèse 4 – *Entre deux corrections successives $(i - 1)$ et i la distribution du temps interdéfaillance X_i connaissant l'instant de défaillance-corrrection T_{i-1} suit une loi exponentielle de paramètre λ_i constant.*

Dans le cadre de l'analyse de la fiabilité des logiciels, le modèle de Jelinski-Moranda [JEL-72] est l'un des premiers modèles étudiés en tant que schéma faisant appel à cette hypothèse. Il est considéré comme étant le modèle le plus simple de la classe des modèles markoviens [XIE-91].

Des modifications et des généralisations ont été apportées au modèle de base de Jelinski-Moranda [MOR-75] [SCH-78] [GOE-80] [TRA-85]. Ainsi, Shanthikumar [SHA-81] a étudié un type de généralisation du modèle de processus de Markov en exprimant l'évolution de la probabilité de transition en fonction du temps.

En 1991, Xie [XIE-91] a passé en revue plusieurs modèles de croissance de fiabilité des logiciels et a proposé une formulation générale du modèle de Markov en supposant que la probabilité de détection des fautes dépend de leur sévérité. Il a montré aussi que cette formulation peut amener à un modèle de croissance de fiabilité général avec des sauts de la fonction intensité de défaillance.

Gaudoin et Soler [GAU-92], pour leur part, ont étudié le modèle proportionnel déterministe en tant que modèle de croissance de fiabilité. Récemment, Soler [SOL-96] a repris l'étude de ce modèle sous des perspectives de corrections différées dans une approche de la croissance de fiabilité des versions d'un logiciel.

Notons que les modèles de croissance de fiabilité peuvent notamment être utilisés dans le cas où la structure du logiciel est introduite. Ainsi, ce type de modèles a été utilisé dans l'analyse structurelle de la fiabilité des logiciels par Littlewood [LIT-79], Cheung [CHE-80], Laprie [LAP-84], les Hecht [HEC-86], Masuda *et al.* [MAS-89] et Ledoux [LED-95].

La formulation que nous proposons dans cet article concerne l'approche boîte noire et n'impose aucune forme paramétrique particulière aux paramètres λ_i et de ce fait généralise plusieurs modèles classiques de la fiabilité des logiciels [HAM-98].

2. Information de Kullback-Leiber et régression antitone

Nous définissons dans ce paragraphe deux notions importantes qui vont nous permettre, par la suite, de déterminer l'estimateur du maximum de vraisemblance non paramétrique de l'intensité de défaillance.

2.1. Information de Kullback-Leiber et estimateur du maximum de vraisemblance

Considérons $(\mathbb{R}, \mathcal{B}(\mathbb{R}), P_\theta)_{\theta \in \Theta}$ un modèle statistique, sur \mathbb{R} , dominé par une mesure μ , S une statistique exhaustive à valeurs dans $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$ et $(x, \theta) \mapsto L(x; \theta)$ une fonction mesurable telle que $L(x; \theta)$ définisse une vraisemblance. Un estimateur du maximum de vraisemblance (EMV) d'un paramètre inconnu θ est une statistique $\hat{\theta}_S$ telle que

$$L(x; \hat{\theta}_S(x)) = \sup_{\theta \in \Theta} L(x; \theta); \quad \forall x \in \mathbb{R} \quad (4)$$

Par ailleurs, l'information de Kullback-Leiber apportée par $P_{\theta'}$ sur P_θ peut être définie, dans ces conditions, par

$$\kappa(\theta, \theta') = \int_{x \in \mathbb{R}} \log \frac{L(x, \theta)}{L(x, \theta')} L(x, \theta) d\mu(x) \quad (5)$$

Ainsi, d'après les relations (4) et (5), il est clair que l'estimateur du maximum de vraisemblance réalise la borne inférieure de l'information de Kullback-Leiber. Dans la suite, l'information de Kullback-Leiber sera considérée comme une mesure de dissemblance entre des probabilités équivalentes.

Maintenant, lorsque $(\mathbb{R}, \mathcal{B}(\mathbb{R}), P_\theta)_{\theta \in \Theta}$ désigne un modèle exponentiel, il existe une application Ψ , de l'ensemble Θ dans \mathbb{R} , liée à la statistique S et à la vraisemblance par

$$L(x; \theta) = \exp(-\Psi(\theta) + \theta S(x))$$

$(\mathbb{R}, \Theta, \Psi)$ définit alors un paramétrage de la famille exponentielle $\{P_\theta\}_{\theta \in \Theta}$. En outre, on montre [DAC-90] que si Ψ prend ses valeurs dans l'intérieur de Θ , elle est C^∞ et dans ces conditions

$$\kappa(\theta, \theta') = \Psi(\theta') - \Psi(\theta) - \Psi'(\theta)(\theta' - \theta). \quad (6)$$

où $\Psi'(\theta)$ désigne la dérivée de $\Psi(\theta)$.

2.2. Principes de la régression antitone

Dans la suite, E désignera un ensemble fini ordonné et $S = (X_u)_{u \in E}$ une séquence de variables aléatoires réelles indicées sur E . Nous dirons que S est une

séquence aléatoire du modèle statistique $(\mathbb{R}, \mathcal{B}(\mathbb{R}), P_\theta)_{\theta \in \Theta}$, s'il existe une unique fonction θ , de l'ensemble $\mathcal{F}(E, \Theta)$ des fonctions de E dans Θ , telle que

$$\forall u \in E : X_u \sim P_{\theta(u)} \quad (7)$$

Ainsi, nous pourrions considérer des séquences aléatoires indicées sur l'ensemble E ou, de manière équivalente, les fonctions correspondantes dans $\mathcal{F}(E, \Theta)$.

Définition 1

Soit $E = \{u_1, u_2, \dots, u_n\}$ un ensemble ordonné d'indices tels que $u_1 \prec u_2 \prec \dots \prec u_n$, une application θ de E vers un intervalle ouvert Θ de \mathbb{R} est dite antitone (ou antitonique), si et seulement si

$$\forall (u, v) \in E \times E : u \prec v \iff \theta(u) \geq \theta(v) \quad (8)$$

Dans cet ordre d'idées, nous désignerons par $\mathcal{A}(E, \Theta)$ l'ensemble des fonctions antitones sur E à valeurs dans l'intervalle Θ . Par ailleurs, soient θ_1 et θ_2 deux fonctions de E dans Θ :

$$(\theta_1, \theta_2) \mapsto d_\omega(\theta_1, \theta_2) = \sqrt{\sum_{u \in E} [\theta_1(u) - \theta_2(u)]^2 \omega(u)}$$

définit une distance euclidienne pondérée sur $\mathcal{F}(E, \Theta)$; où $\omega(u)$ est une pondération strictement positive.

Définition 2

Soient, θ une fonction de E vers Θ et ω une fonction positive sur E . On appelle régression antitone de θ une application θ^\dagger qui minimise la distance euclidienne pondérée entre θ et la classe $\mathcal{A}(E, \Theta)$. Autrement dit

$$\theta^\dagger = \arg \min_{\tilde{\theta} \in \mathcal{A}(E, \Theta)} d_\omega(\tilde{\theta}, \theta). \quad (9)$$

Nous montrons dans [HAM-98] [HAM-99b] que la régression antitone de θ peut se mettre sous la forme

$$\theta^\dagger(u_i) = \min_{s \leq i} \max_{r \geq s} \frac{\sum_{k=s}^r \theta(u_k) \omega(u_k)}{\sum_{k=s}^r \omega(u_k)} \quad \text{pour } i = 1..n. \quad (10)$$

2.3. Théorèmes d'antitonsation

L'objectif de ce paragraphe est d'établir le cadre formel de l'équivalence entre le problème de régression antitone et le problème de l'estimation du maximum de vraisemblance sous des restrictions d'ordre.

Définition 3

Soit E un ensemble fini et ordonné d'indices pondérés par un poids positif ω . Considérons θ_1 et θ_2 deux fonctions de $\mathcal{F}(E, \Theta)$, nous définissons alors l'information de Kullback-Leiber de la fonction θ_1 sur la fonction θ_2 comme étant l'information de Kullback-Leiber cumulée des valeurs prises par θ_1 sur les valeurs prises par θ_2 i.e.

$$\mathcal{K}_\omega(\theta_1, \theta_2) = \sum_{u \in E} \kappa[\theta_1(u), \theta_2(u)] \omega(u) \quad (11)$$

Nous démontrons dans [HAM-99b] les résultats suivants :

Proposition 1

Soit E un ensemble fini et ordonné d'indices pondérés par un poids positif ω et $\{P_\theta\}_{\theta \in \Theta}$ une famille exponentielle telle que la variété des paramètres Θ soit un ensemble convexe de \mathbb{R} . Considérons θ une fonction sur $\mathcal{F}(E, \Theta)$ dont l'ensemble des valeurs s'inscrit à «l'intérieur» de l'ensemble Θ . Alors, la régression antitone θ^\dagger de θ minimise l'information de Kullback-Leiber $\mathcal{K}_\omega(\tilde{\theta}, \theta)$ sur la classe $\mathcal{A}(E, \Theta)$. Autrement dit

$$\theta^\dagger \in \text{Arg} \min_{\tilde{\theta} \in \mathcal{A}(E, \Theta)} \mathcal{K}_\omega(\tilde{\theta}, \theta) \quad (12)$$

Si, par ailleurs, la fonction Ψ caractérisant le paramétrage de la famille exponentielle $\{P_\theta\}_{\theta \in \Theta}$ est strictement convexe, alors le minimum est unique, i.e.

$$\theta^\dagger = \arg \min_{\tilde{\theta} \in \mathcal{A}(E, \Theta)} \mathcal{K}_\omega(\tilde{\theta}, \theta) \quad (13)$$

Corollaire 2 – Dans les conditions de la proposition 1 et avec les notations ci-dessus, la régression antitone θ^\dagger de θ maximise la quantité

$$\sum_{u \in E} \{\Psi \circ \tilde{\theta}(u) + [\theta(u) - \tilde{\theta}(u)](\Psi' \circ \tilde{\theta})(u)\} \omega(u) \quad (14)$$

Démonstration :

Le corollaire découle directement de la proposition 1 et du fait que d'après les relations (6) et (11), l'expression (14) est égale à

$$\sum_{u \in E} \Psi \circ \theta(u) \omega(u) - \mathcal{K}_\omega(\tilde{\theta}, \theta)$$

3. Application du modèle à l'évaluation de la fiabilité

3.1. EMV de l'intensité de défaillance

Soit $N_t = n$, le nombre des défaillances observées jusqu'à l'instant t , $E = \{1, 2, \dots, n\}$ et $t_1 < t_2 < \dots < t_n$ les instants de défaillance correspondants (avec $t_0 = 0 < t_1$); d'après les hypothèses 3 et 4, l'intensité de défaillance est une fonction étagée décroissante de $\mathcal{A}(E, \mathbb{R})$ demeurant constante entre deux défaillances-corrrections successives. Aussi, l'équation (1) permet d'écrire

$$\forall x \geq 0 : R_i(x) \stackrel{\text{déf}}{=} R_{t_i}(x) = \exp(-\lambda_i x), \text{ pour } i = 1..n \quad (15)$$

et le problème d'estimation du maximum de vraisemblance du paramètre $\lambda_{(\cdot)}$ qui caractérise l'histoire des corrections apportées au logiciel aux instants t_1, t_2, \dots et t_n , consiste à maximiser la fonction vraisemblance aux points $x_i = t_i - t_{i-1}$ pour $i = 1..n$:

$$L(x_1, x_2, \dots, x_n; \lambda_{(\cdot)}) = \prod_{i=1}^n f_{X_i/T_{i-1}=t_{i-1}}(x_i) = \prod_{i=1}^n \lambda_i \exp(-\lambda_i x_i) \quad (16)$$

sous la contrainte $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. Dans ces conditions, le problème d'optimisation pour la fonction log-vraisemblance prend la forme

$$\begin{cases} \text{Maximiser } \sum_{i=1}^n [x_i^{-1} \log \lambda_i - \lambda_i] x_i \\ \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \end{cases} \quad (17)$$

Maintenant, en considérant la fonction convexe $\Psi(u) = u \log u - u$, en posant $\omega(i) = x_i$, $\theta(i) = x_i^{-1}$ et $\tilde{\theta}(i) = \lambda_i$ pour $i = 1..n$ et en prolongeant l'expression (6) établie pour l'information de Kullback-Leiber à l'ensemble des fonctions convexes, les résultats énoncés dans le paragraphe 2.3. restent applicables et on reconnaît en (17) une forme du corollaire 2 avec la fonction Ψ que nous avons adoptée.

D'après la relation (10), la solution du problème d'optimisation (17) est en conséquence

$$\lambda_{i:n}^\dagger = \min_{s \leq i} \max_{r \geq s} \frac{r - s + 1}{r \sum_{j=s}^r (t_j - t_{j-1})} = \min_{s \leq i} \max_{r \geq s} \frac{r - s + 1}{t_r - t_{s-1}} \text{ pour } i = 1..n \quad (18)$$

Ainsi, l'estimateur du maximum de vraisemblance, sous la restriction d'antitonicité de l'intensité de défaillance, basé sur la séquence d'observations $\{t_i/i = 1..n\}$ est

$$\lambda_{(\cdot)}^\dagger = (\lambda_{1:n}^\dagger, \lambda_{2:n}^\dagger, \dots, \lambda_{n:n}^\dagger). \quad (19)$$

Toutefois, il faut remarquer qu'avec la fonction convexe Ψ adoptée dans ce paragraphe, l'expression (6) associée ne correspond plus à l'information de Kullback-Leiber, mais les valeurs λ_i maximisant (17) minimisent cette information.

3.2. Aptitudes prédictives du modèle

La fiabilité d'un logiciel est souvent estimée et prédite en analysant les données des défaillances collectées par le passé. En se fixant une origine des temps, les données que nous considérons ici sont les instants de défaillance lorsque le logiciel est en train de subir des corrections durant la phase de test ou opérationnelle de son cycle de vie. Nous considérons aussi une forme de prédiction séquentielle d'ordre 1; c'est-à-dire qu'à partir de l'ensemble des données $\{t_i/i = 1..n\}$, nous essaierons de prédire le comportement du temps T_{n+1} de la prochaine défaillance.

Remarquons tout d'abord que l'estimateur décrit par les relations (18) et (19) n'est déterminé de manière unique que sur l'intervalle de temps $[0, T_n[$. La fonction étagée, intensité de défaillance estimée, correspondante peut donc être prolongée de la manière suivante

$$\forall t > 0 : \hat{\lambda}_t = \begin{cases} \lambda_{i:n}^\dagger & \text{si } T_{i-1} \leq t < T_i \text{ pour } i = 1..n \\ c_n & \text{si } t \geq T_n \end{cases}$$

où c_n est une constante arbitrairement choisie telle que $c_n \leq \lambda_{n:n}^\dagger$ pour préserver l'hypothèse 3. En effet, tant que nous n'avons pas encore observé une nouvelle défaillance, nous ne pouvons avoir une meilleure information *a priori*. Plusieurs méthodes statistiques de prévision peuvent servir d'aide au choix pour déterminer la constante c_n au vu d'un échantillon donné. Dans une approche de prédiction «*au pire*» nous proposons de fixer : $c_n = \lambda_{n:n}^\dagger$; cela reviendrait à dire : «*puisque'il s'agit d'un modèle de croissance de fiabilité, au pire des cas c_n vaudrait $\lambda_{n:n}^\dagger$* ». C'est le cas, par exemple, lorsque la tentative d'élimination des fautes qui ont causé la $n^{\text{ième}}$ défaillance n'a pas abouti.

La fiabilité prédictive après n défaillances observées vaut :

$$\hat{R}_{n+1:n}(t) = P(X_{n+1} \geq t \mid T_1 = t_1, \dots, T_n = t_n) = \exp(-\lambda_{n:n}^\dagger t)$$

et la durée moyenne prédictive de bon fonctionnement du logiciel, connaissant les instants de défaillance $\{t_i/i = 1..n\}$, par analogie avec la relation (2), peut être estimée dans le cas du présent modèle par :

$$\widehat{MTTF}_n = E(X_{n+1} \mid T_1 = t_1, \dots, T_n = t_n) = \int_0^{+\infty} \hat{R}_{n+1:n}(t) dt = \frac{1}{\lambda_{n:n}^\dagger}. \quad (20)$$

En outre, les performances prédictives du modèle sur des jeux de données réelles peuvent être évaluées, par exemple, à l'aide des tests d'adéquation. Pour notre part, nous utiliserons ici la méthode du u-plot [COX-66] [ABD-86].

3.3. Méthode du u-plot

La méthode du u-plot définit un indicateur d'adéquation non-paramétrique [COX-66]. L'objectif de cette procédure est de déterminer si les prédictions $\{\hat{F}_{n:i}\}_{i=s..r}$ sont assez proches des distributions réelles $\{F_{n:i}\}_{i=s..r}$ inconnues des X_i (avec $1 \leq s \leq i \leq r \leq n+1$). Si ces distributions étaient identiques alors, cela signifierait que les variables aléatoires $U_i = \hat{F}_{n:i}(X_i)$ sont uniformément distribuées sur l'intervalle $]0, 1[$. Ainsi, soient $(x_i)_{i=s..r}$ le vecteur des réalisations des temps interdéfaillances $(X_i)_{i=s..r}$; les $u_i = \hat{F}_{n:i}(x_i)$ sont des réalisations des v.a. U_i . Le vecteur ordonné $(u_{(j)})_{j=1..r-s+1}$ constitue une réalisation du vecteur des statistiques d'ordre $(U_{(j)})_{j=1..r-s+1}$ associé au vecteur $(U_i)_{i=s..r}$.

La courbe du u-plot est alors la fonction de répartition empirique des $u_{(j)}$. Elle est définie par la fonction étagée, à pas constant, qui à $u_{(j)}$ associe $\frac{j}{r-s+1}$ (pour $j = 1, \dots, r-s+1$). Cette courbe permet de constater visuellement la performance du modèle dans l'interprétation des données observées.

Les statistiques de Kolmogorov-Smirnov

$$\begin{cases} D_{s,r}^+ &= \max_{1 \leq j \leq r-s+1} \left(\frac{j}{r-s+1} - U_{(j)} \right) \\ D_{s,r}^- &= \max_{1 \leq j \leq r-s+1} \left(U_{(j)} - \frac{j-1}{r-s+1} \right) \\ KS_{s,r} &= \max(D_{s,r}^-, D_{s,r}^+) \end{cases} \quad (21)$$

permettent de définir un test non-paramétrique bilatéral

$$H_0 : X_i \sim F_{n:i} \text{ pour } i = s, \dots, r \text{ contre } H_1 : \exists i_0 \in [s, r] \text{ tel que } X_{i_0} \not\sim F_{n:i_0},$$

qui rejette l'hypothèse nulle au seuil α , si on observe $KS_{s,r} > d_{s,r}^\alpha$; où $d_{s,r}^\alpha$ est la valeur lue sur la table de Kolmogorov-Smirnov à $(r-s+1)$ degrés de liberté pour le seuil de risque α .

Par ailleurs et de manière à quantifier l'aptitude du modèle à interpréter les données $(X_i)_{i=s..r}$, nous pouvons définir [HAM-99a] deux indicateurs induits par le u-plot du modèle proposé,

- la statistique définissant l'écart moyen empirique associé au u-plot du modèle

$$EMGA_{s,r} = \frac{1}{r-s+1} \sum_{j=1}^{r-s+1} \left| \frac{j}{r-s+1} - U_{(j)} \right| \quad (22)$$

- la statistique de l'erreur quadratique moyenne empirique incitée par le u-plot du modèle

$$EMSE_{s,r} = \sqrt{\frac{1}{r-s+1} \sum_{j=1}^{r-s+1} \left(\frac{j}{r-s+1} - U_{(j)} \right)^2} \quad (23)$$

3.4. Analyse des données de défaillance-correction

Cette analyse a été réalisée à l'aide de l'outil COALAS [HAM-98] [HAM-99a]. C'est un système dédié au traitement formel des applications statistiques. Le système COALAS regroupe un ensemble de bibliothèques statistiques et graphiques permettant l'analyse paramétrique et non-paramétrique de la fiabilité des logiciels, à la fois, par des approches dynamiques et statiques. Nous l'avons écrit autour du noyau du système de calcul formel MAPLE V.

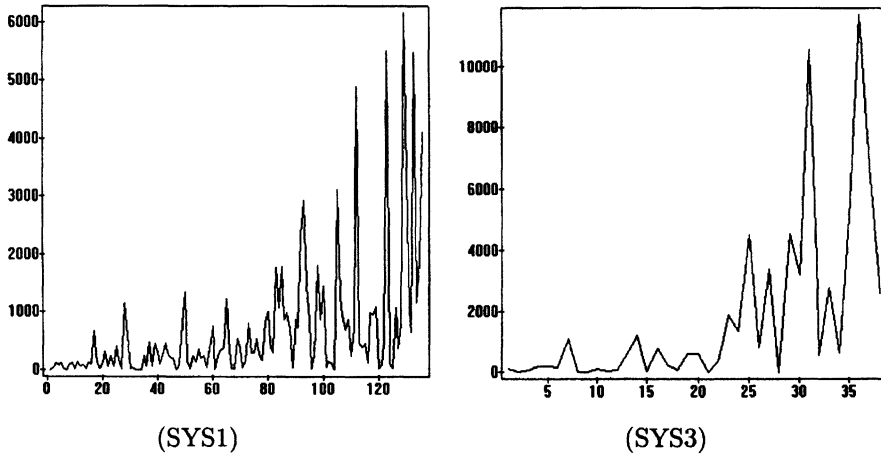


FIGURE 1

Temps interdéfaillance des systèmes SYS1 et SYS3 (voir le paragraphe 1.2)

L'objectif de ce paragraphe est d'analyser des données de défaillance-correction émanant de véritables projets de fiabilité à l'aide des méthodes établies dans cet article et de comparer les performances de notre modèle, identifié dans la suite par l'attribut NOPRA, à celles des modèles résumés par le tableau suivant :

Symbole	Signifié et référence de base	Intensité de défaillance
JM	modèle de Jelinski-Moranda [JEL-72]	$\lambda_i = \phi [N - (i - 1)]$
GM	modèle géométrique de Moranda [MOR-75]	$\lambda_i = \lambda_1 \exp[-\theta (i - 1)]$
YOO	modèle de Yamada-Ohba-Osaki [YAM-83]	$\lambda_t = \alpha \phi^2 \exp(-\phi t) t$
GO	modèle de Goel-Okumoto [GOE-80]	$\lambda_t = \alpha \phi \exp(-\phi t)$
PCD	modèle puissance de Crow-Duane [CRO-74]	$\lambda_t = \alpha \beta t^{\beta-1}$
NOPRA	modèle non-paramétrique [HAM-98]	$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$

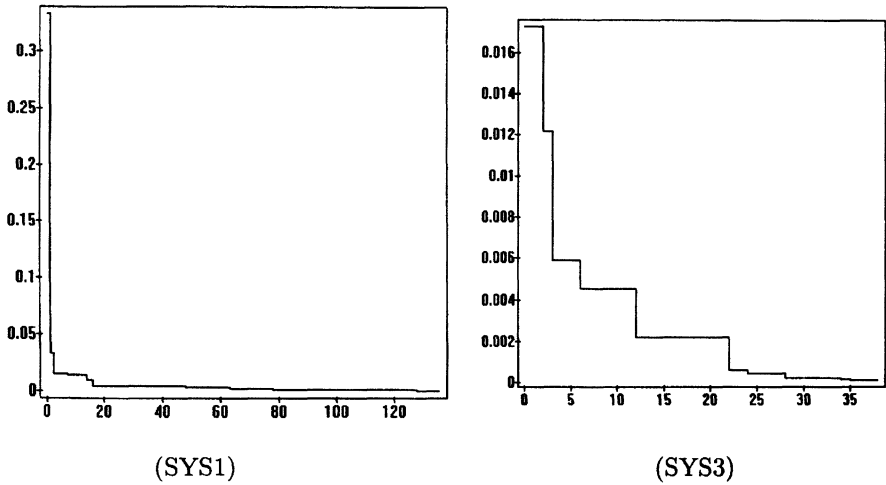


FIGURE 2
Intensités de défaillances des deux systèmes (voir la formule (18))

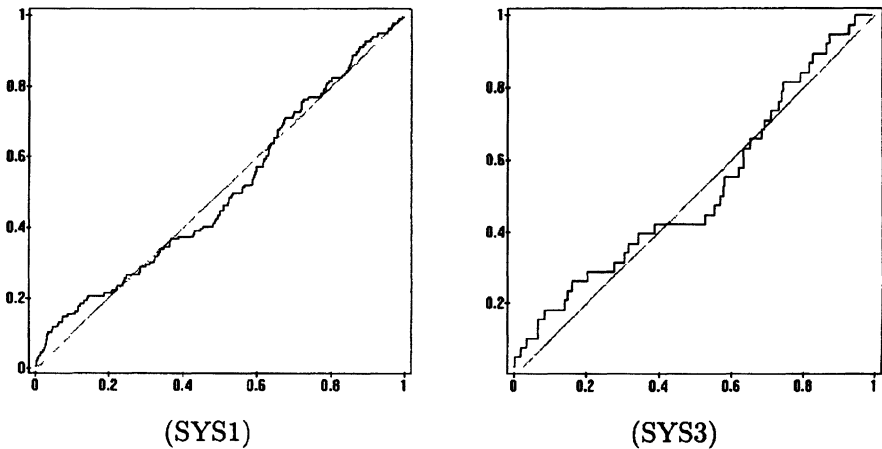


FIGURE 3
Courbes des u-plots prédictifs du modèle NOPRA pour les deux systèmes
(voir le paragraphe 3.3)

Les figures 1 (SYS1) et 1 (SYS3) montrent que les données récoltées, lors du test fonctionnel des produits logiciels SYS1 et SYS3 [MUS-79], présentent globalement une croissance de fiabilité. En effet, les intervalles des temps interdéfaillances sont globalement de plus en plus grands; ce qui justifie l'application d'un modèle de croissance de fiabilité.

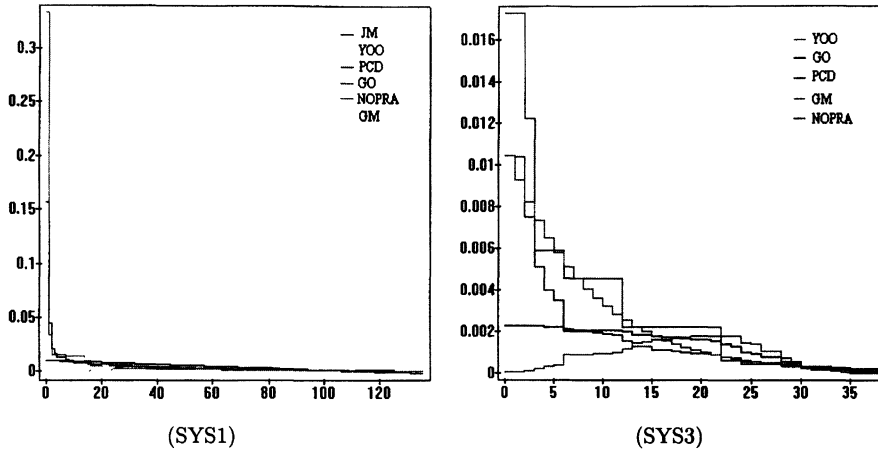


FIGURE 4

*Comparaison des intensités de défaillance pour les différents modèles
(voir le tableau des symboles du paragraphe 3.4)*

Selon le modèle NOPRA, la figure 2 (SYS1) montre que les fautes les plus sévères détectées par le plan de test considéré ont été éliminées au début du test. Cette figure montre aussi que leur nombre n'est pas très élevé (une demi-douzaine), à moins que le plan de test du logiciel n'ait pas permis de révéler, assez tôt, toutes les fautes sévères au début de la phase du test.

La même remarque peut être faite à propos de la figure 2 (SYS3), sauf que pour le système SYS3 la détection des fautes sévères a continué jusqu'à un stade assez avancé du test du logiciel puisqu'on observe des sauts de forte amplitude au voisinage de la 22^{ème} défaillance. Des interprétations différentes du plan de test peuvent être produites, si on se fie aux allures des courbes de l'intensité de défaillance des autres modèles décrits dans les figures 4 (SYS1), 4 (SYS3), 5 (SYS1) et 5 (SYS3) où l'intensité de défaillance demeure très faible comparée à l'intensité de défaillance du modèle NOPRA au début du test.

Dans les tableaux 1 et 2, nous introduisons l'indicateur $KL_{s,r}$ désignant la mesure de dissemblance de Kullback-Leiber cumulée et normée associée aux poids $\omega_i = x_i$ pour $i = s, \dots, r$ du modèle considéré par rapport au modèle non paramétrique NOPRA. Ainsi, comme les intensités de défaillances utilisées sont estimées par des fonctions étagées, à savoir, λ pour le modèle considéré et λ^\dagger pour le modèle NOPRA, alors $KL_{s,r}$, dans ces conditions, est donnée par l'expression

$$KL_{s,r} = \frac{1}{r} \sum_{i=s}^r \left[\log \frac{\lambda_i^\dagger}{\lambda_i} + \frac{\lambda_i}{\lambda_i^\dagger} \right] x_i - 1 \quad (24)$$

En somme, les mesures exposées dans les tableaux 1 et 2 et les figures 4 (SYS1), 4 (SYS3), 5 (SYS1) et 5 (SYS3), pour les systèmes SYS1 et SYS3,

TABLEAU 1
*Mesures relatives à la comparaison des modèles pour le système SYS1
 (voir les formules (3), (21), (22), (23) et (24))*

Modèle	$MTTF_{137}$	$KS_{6,106}$	$EMGA_{6,106}$	$EMSE_{6,106}$	$KL_{1,136}$
JM	$+\infty$	0.21980	0.11217	0.01283	$+\infty$
GM	2295.49534	0.09222	0.02844	0.00370	0.04095
YOO	$+\infty$	0.19484	0.07695	0.00934	5.74737
GO	$+\infty$	0.10103	0.03664	0.00438	0.05541
PCD	1429.00000	0.10139	0.04997	0.00548	0.08621
NOPRA	4116.00000	0.10035	0.03287	0.00421	*

TABLEAU 2
*Mesures relatives à la comparaison des modèles pour le système SYS3
 (voir les formules (3), (21), (22), (23) et (24))*

Modèle	$MTTF_{38}$	$KS_{6,35}$	$EMGA_{6,35}$	$EMSE_{6,35}$	$KL_{1,38}$
JM	–	–	–	–	–
GM	8535.21614	0.15215	0.05850	0.01353	0.02191
YOO	$+\infty$	0.28834	0.09383	0.02285	3.19746
GO	$+\infty$	0.19771	0.06996	0.01629	0.27795
PCD	4629.58794	0.18709	0.08469	0.01813	0.08683
NOPRA	6988.66667	0.10579	0.04184	0.00913	*

* Une donnée évidemment nulle.

– Le calcul des estimateurs des paramètres a divergé.

relatent une meilleure adéquation du modèle paramétrique GM pour l'interprétation des données de défaillance-correction de SYS1 et du modèle non-paramétrique NOPRA pour l'interprétation des données du système SYS3. Cela étant, il faut noter que l'information de Kullback-Leiber cumulée $KL_{s,r}$ des différents modèles paramétriques relativement au modèle non-paramétrique NOPRA, indique dans les deux cas une plus faible dissemblance pour le modèle paramétrique GM.

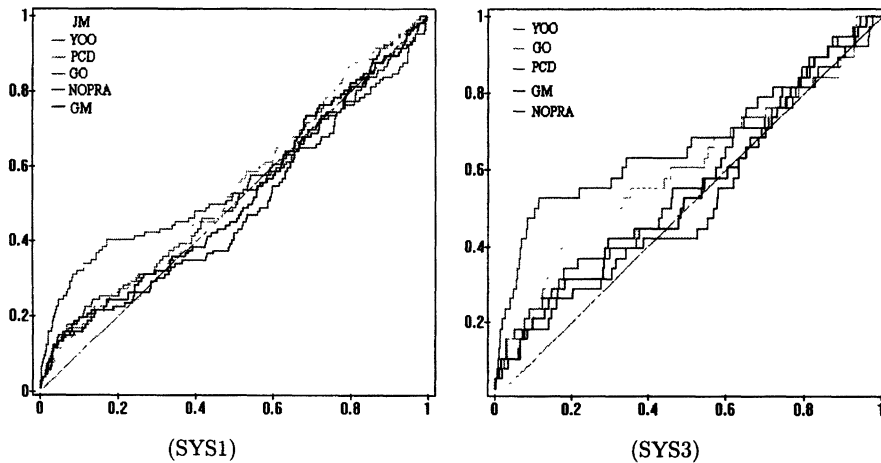


FIGURE 5
Confrontation des u-plots pour les différents modèles

4. Conclusions

Dans cet article, nous avons présenté une approche d'estimation et de prédiction de la fiabilité des logiciels basée sur la méthode du maximum de vraisemblance sous l'hypothèse de la croissance de fiabilité. Par ailleurs, nous avons défini deux indicateurs de la qualité prédictive des modèles basés sur la courbe du u-plot; indicateurs qui avec la distance de Kolmogorov-Smirnov nous ont permis de donner une idée sur la performance des modèles et d'offrir un instrument pour le choix du modèle le plus adéquat à l'interprétation des données d'un projet de fiabilité.

L'approche d'estimation et de prédiction de la croissance de fiabilité des logiciels que nous avons proposée, nous semble être prometteuse. Son intérêt majeur réside dans le fait qu'elle ne restreint pas l'intensité de défaillance à une quelconque forme analytique. Elle permet ainsi de débarrasser la modélisation de la croissance de fiabilité de bien de contraintes inutiles. Elle peut être généralisée à la modélisation et à l'étude de la croissance de fiabilité d'une grande classe de systèmes améliorables.

Remerciements

Je tiens à remercier les relecteurs. La pertinence de leurs remarques m'a permis d'améliorer différents points de ce texte.

Bibliographie

- [ABD-86] ABDEL-GHALY A.A., CHAN P.Y., LITTLEWOOD B. (1986), Evaluation of competing software reliability predictions, IEEE Trans. on Soft. Eng., SE-12 (9). pp. 950-967.

- [BAR-71] BARLOW R.E., UBHAYA V.A. (1971), Isotonic Approximation. In Optimizing methods in statistics, J.S. Rustagi ed., New York.
- [CHE-80] CHEUNG R.C. (1980), A user oriented reliability model, IEEE Trans. Software Eng., SE-6, pp. 118-125.
- [COU-73] COUTINHO J. De H. (1973), Software reliability growth, Record of 1973 IEEE Symp. on Comp. Soft. Rel. IEEE Computer Society, NY, pp. 58-64.
- [COX-66] COX D.R., LEWIS P.A.W. (1966), Statistical analysis of series of events, Methuen Monographs on Applied Probability and Statistics, London, Methuen & Co Ltd.
- [CRO-74] CROW L.H. (1974), Reliability analysis for complex repairable systems, Reliability and Biometry, Statistical analysis of lifelength, SIAM Philadelphia, pp. 379-410.
- [DAC-90] DACUNHA-CASTELLE D., DUFLO M. (1990), Probabilités et statistiques, Tome 1, Masson.
- [GAU-92] GAUDOIN O., SOLER J.L. (1992), Statistical analysis of the geometric de-eutrophication software reliability model, IEEE Trans. on Reliability, R-41(4), pp. 518-524.
- [GOE-80] GOEL A.L. (1980), A summary of the discussion on an analysis of competing software reliability models, IEEE Trans. Software Eng., SE-6, pp. 501-502.
- [HAM-98] HAMLILI A. (1998), Estimation de la fiabilité des logiciels sous hypothèse de leur amélioration, Accepté pour publication. Actes de la deuxième conférence internationale sur les mathématiques appliquées et les sciences de l'ingénieur (CIMASI'98), pp. 232-237, Casablanca, Maroc, 27-29 Oct.
- [HAM-99a] HAMLILI A. (1999), Etude comparative des modèles de croissance de fiabilité des logiciels, Actes de la deuxième conférence internationale en recherche opérationnelle, Marrakech, Maroc, 24-26 Mai.
- [HAM-99b] HAMLILI A. (1999), Contribution à l'évaluation et la prédiction de la fiabilité des logiciels : méthodologies, modèles, inférence statistique et qualité. Thèse de doctorat d'état ès sciences appliquées. Université Mohamed V, Ecole Mohammedia d'Ingénieurs, (soutenance public le 8 septembre 1999).
- [HEC-86] HECHT H., HECHT M. (1986), Software reliability in the system context, IEEE Trans. Software Eng., SE-12, pp. 51-58.
- [IEEE94] IEEE Standards Collection (1994), Software Engineering, published by the Institute of Electrical and Electronic Engineers Inc..
- [ISO-98] ISO 9000-3 (1998), Normes pour le management de la qualité et l'assurance de la qualité, Partie 3 : Lignes directrices pour l'application de l'ISO 9001 :1994 au développement, à la mise à disposition, à l'installation et à la maintenance du logiciel, Organisation internationale de normalisation, Suisse, version française.

- [JEL-72] JELINSKI Z., MORANDA P.B. (1972), Software reliability research, in Stat. Comp. Performance Evaluation. Ed. Freiberber W. pp. 465-484, Academic NY.
- [LAP-84] LAPRIE J.C. (1984), Dependability evaluation of software systems in operation, IEEE Trans. Software Eng., SE-16(6), pp. 701-714.
- [LED-95] LEDOUX J. (1995), Sur la modélisation structurelle markovienne en fiabilité des logiciels. Rapport INRIA, n° 2714.
- [LIT-73] LITTLEWOOD B., VERRALL J.L. (1973), A bayesian reliability growth model for computer software, Jour. Roy. Stat. Soc., C vol. 22, pp. 332-346.
- [LIT-79] LITTLEWOOD B. (1979), Software reliability model for modular program structure, IEEE Trans. Reliability, R-28, pp. 241-246.
- [LYU-96] LYU M.R. (1996), Handbook of software engineering, Computing McGraw-Hill, New York.
- [MAS-89] MASUDA Y. *and al.* (1989), A statistical approach for determining time of software system with modular structure, IEEE Trans, Reliability, R-38, pp. 365-372.
- [MOR-72] MORANDA P.L., JELINSKI Z. (1972), Final report on software reliability study, McDonnell Douglas Astronautics Compagny, MADC report number 63921.
- [MOR-75] MORANDA P.B. (1975), Predictions of software reliability during debugging. Proc. of the Annual Reliability and Maintainability Symposium, pp. 327-332, Washinton, D.C., USA.
- [MUS-79] MUSA J.D. (1979), Software reliability data. Report available from data and analysis center for software, Rome air development center, New York.
- [MUS-87] MUSA J.D, IANNIO A., OKUMUTO A. (1987), Software reliability : measurement, prediction, application, McGraw-Hill, New York.
- [SCH-78] SCHICK G.J., WOLVERTON R.W. (1978), An analysis of competing software reliability models, IEEE Trans. Software Eng. SE-4, pp. 104-120.
- [SHA-81] SHANTHIKUMAR J.G. (1981), A general software reliability model for performance prediction, Microelectronics and reliability, vol. 21, pp. 671-682.
- [SHO-72] SHOOMAN M.L. (1972), Probabilistic models for software reliability predictions, in Stat. Comp. Performance Evaluation, ed. Freiberber W., pp. 485-502, Academic press, NY.
- [SHO-84] SHOOMAN M.L. (1984), Software reliability : a historical perspective, IEEE Trans. Software Eng. R-33, pp. 48-55.
- [SNY-75] SNYDER D.L. (1975), Random point processes, Wiley, New York.
- [SOL-96] SOLER J.L. (1996), Croissance de fiabilité des versions d'un logiciel. Rev. Stat. App., XLIV(1), pp. 5-20.

- [TRA-85] TRACHTENBERG M. (1985), The linear software reliability model and uniform testing, IEEE Trans. Reliability, R-34, pp. 8-16.
- [XIE-91] XIE M. (1991), Software reliability modelling, World Scientific, Singapore.
- [YAM-83] YAMADA S., OHBA M., OSAKI S. (Dec. 1983), S-Shaped reliability growth modeling for software error detection, IEEE Trans. on Rel., vol. R-32, pp. 475-478.