Analyse des défaillances et modélisation du comportement du logiciel d'un autocommutateur téléphonique.

Thierry Sabourin, Karama Kanoun

ANALYSE DES DEFAILLANCES ET MODELISATION DU COMPORTEMENT DU LOGICIEL D'UN AUTOCOMMUTATEUR TELEPHONIQUE

FAILURE ANALYSIS AND MODELISATION OF SOFTWARE BEHAVIOUR IN ELECTRONIC A SWITCHING SYSTEM

T. SABOURIN
Alcatel Commutation B.P.344
22304 LANNION (France)

K. KANOUN
LAAS/CNRS 7, Av. du col. Roche
31400 TOULOUSE (France)

RESUME

Cet article traite du problème de l'évaluation de la fiabilité du logiciel durant les phases de validation et de vie opérationnelle. L'étude est basée sur l'observation et l'analyse des défaillances, et des corrections associées, d'un important système temps-réel. Dans un premier temps le logiciel est considéré comme une boite noire, cette démarche est affinée en considérant à la fois les conséquences des défaillances et la structure du logiciel. A chaque étape une vérification expérimentale a été effectuée.

ABSTRACT

This paper deals with the evaluation of the reliability of the software during the validation and operational phases. The study is based on the analysis of observed failures and corresponding corrections of a large real-time system. First, the software is viewed as a black-box, then failure consequences and the software structure are considered. An experimental verification has been carried out at each step.

INTRODUCTION

Il est inutile de rappeler l'importance d'une étude prévisionnelle de la sûreté de fonctionnement dès la conception d'un système, surtout si le système en question est soumis à des exigences de qualité de service très sévères.

Ce document présente une analyse des défaillances logicielles survenues sur un sous-ensemble d'autocommutateur téléphonique. Le sous-ensemble a été suivi depuis la phase de validation jusqu'à une période avancée de la vie opérationnelle.

La première partie présente le système étudié ainsi que la saisie des informations nécessaires à l'analyse. Une analyse préliminaire nous permet, parallèlement à l'application de modèles mathématiques, de mieux comprendre le déroulement des phases de validation et de vie opérationnelle.

La deuxième partie est consacrée à une étude qualitative puis quantitative dans le temps des défaillances et des corrections associées. Pour cela on réalise des répartitions de ces défaillances selon les différents composants logiciels concernés et selon les différentes conséquences.

Dans la troisième partie on teste la croissance de fiabilité, puis on applique les modèles de croissance de fiabilité les plus couramment utilisés. Cela permet d'évaluer la fiabilité du logiciel du système tout au long de la période considérée. On observe ainsi sa croissance de fiabilité puis sa fiabilité en régime opérationnel stabilisé.

Les bases d'un modèle de fiabilité de logiciel sont ensuite vérifiées; ce modèle tient compte du composant concerné, de la conséquence de la défaillance et du taux d'activation de ces différents composants.

La terminologie utilisée est celle de (Lap 85).

I) DESCRIPTION DU SYSTEME ET MODE DE COLLECTE DE DONNEES

I.1) DESCRIPTION DU SYSTEME ETUDIE

Le système objet de l'étude est un sous-ensemble d'un système de commutation. Il assure essentiellement des fonctions de commutation, d'exploitation, de surveillance du bon fonctionnement du système et de localisation des fautes matérielles.

Afin d'assurer une bonne disponibilité le système est redondant (figure 1), il est composé de deux unités appelées "LOGIQUES" parfaitement similaires. Elles travaillent en un mode de redondance dynamique partielle: une unité "pilote" remplit les fonctions du système, l'autre dite "réserve" met à jour ses propres tables à partir de celles de l'unité pilote de manière à devenir active à la place de l'autre en cas de défaillance de cette dernière.
Le système comprend de plus un module de choix d'unité qui contient très peu d'éléments et qui peut être activé soit par l'opérateur soit par l'unité "pilote".

Tous les processeurs sont identiques au niveau matériel.

Chaque unité est constituée de trois processeurs se situant à différents niveaux fonctionnels. Un processeur principal gère les échanges entre les différents processeurs et les autres organes de l'autocommutateur, réalise des fonctions de surveillance et de contrôle des tâches, il organise la maintenance et l'exploitation de l'organe. Deux processeurs auxiliaires assurent les fonctions de commutation et d'échange avec le processeur principal. Les fonctions de détection des défaillances sont réalisées au niveau de chaque processeur, les procédures de recouvrement étant mises en oeuvre par les processeurs principaux.

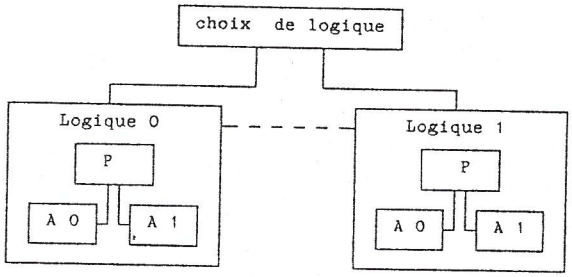

Figure 1: Structure du système

En absence de défaillance, une unité est constituée d'une copie du logiciel de type "principal" et de deux copies du logiciel de type "auxiliaire".

I.2)TEST-VALIDATION ET FICHES DE SUIVI CORRESPONDANTES

Le Test

Le test du logiciel est réalisé par les personnes ayant écrit le code. Il est effectué en deux étapes:
- le Test Unitaire effectué sur une machine de simulation de l'environnement, les corrections étant directement apportées par le programmeur,
- le Test Maquette effectué par fonction du système lors de l'intégration, chaque modification donne lieu à la rédaction d'un Ordre de Changement (OC) dans le logiciel.

La Validation

Lors de la validation le système est placé dans des configurations semblables à celles de la vie opérationnelle. Une équipe indépendante de l'équipe de développement élabore des fiches d'essais et effectue la validation. Si le test de validation est positif, la fiche d'essais est considérée comme validée, dans le cas contraire un Relevé Technique d'Anomalie (RTA) est rédigé (une fiche d'essais peut donner lieu à plusieurs RTA). Ce RTA entraine une correction par l'équipe de développement qui, en retour, génère une nouvelle version. On effectue alors, à nouveau, le test qui a activé l'erreur, afin de valider la fiche d'essais concernée.

On peut ainsi distinguer trois phases dans le

travail de validation: le test, l'identification de l'erreur et l'élimination de cette erreur.

Après la correction, les traces de l'anomalie rencontrée sont:
- le RTA qui décrit précisément les conditions d'apparition de la défaillance et les conséquences sur le fonctionnement du système,
- l'OC qui décrit les raisons techniques de la défaillance et mentionne le (ou les) composant(s) concerné(s) ainsi que les corrections effectuées.

La vie opérationnelle

Au début de la vie opérationnelle les fautes sont découvertes soit par l'équipe de validation qui continue à travailler sur le système, soit sur les sites d'exploitation. Dans tous les cas on peut retrouver les RTA écrits au moment de la simulation de l'anomalie sur maquette et les OC écrits lors de la correction.

II) ANALYSE DES ANOMALIES LOGICIELLES

II.1) ANALYSES PRELIMINAIRES

II.1.1) Les données relatives à la fiabilité extraites des feuilles de suivi du logiciel

On extrait des fiches de relevé technique d'anomalie les intervalles de temps entre défaillances. Ces intervalles ne sont pas rigoureusement des durées entre défaillances du système mais des intervalles entre rédactions de RTA.

On a vu plus haut que les anomalies découvertes en interne n'entrainaient pas de création de RTA mais uniquement d'OC. De ces OC on extrait les intervalles entre corrections. Ils permettent d'identifier les composants concernés et les types des fautes.

II.1.1.1) Conditions d'apparition des anomalies

Les conditions d'apparition rencontrées ont été réparties en trois classes selon les différents types de sollicitations auxquelles le système est soumis:
- .**cas A**: la défaillance est apparue en fonctionnement de téléphonie, cas où le système ne fait ni commande opérateur, ni traitement après défaillance matérielle ou après basculement de logique,
- .**cas B**: la défaillance logicielle est apparue suite à une commande opérateur alors que le système est exempt de défaillance matérielle et qu'aucun basculement n'a lieu durant l'exécution de la commande,
- .**cas C**: apparition de la défaillance suite à une combinaison d'événements: regroupe tous les cas de manifestation de fautes à l'exeption des cas A et B.

II.1.1.2) Conséquences des anomalies

Les conséquences des anomalies ont été réparties en six groupes correspondant aux conséquences les plus fréquemment rencontrées:
- .**INDisponibilité GEnérale** (INDGE): résulte soit de la défaillance non détectée du pilote, soit d'une défaillance grave entrainant une RAZ du système,
- .**PERte d'une LOgique** (PERLO): cas correspondant aux défaillances d'une logique (matérielle ou logicielle) entrainant un basculement du système qui continue à travailler normalement,
- .**Abandon d'un Traitement d'EXPloitation** (ATEXP): cas d'une commande opérateur non exécutée.
- .**INDisponibilité de CIrcuit** (INDCI): cas correspondant aux défaillances entrainant l'indisponibilité d'un (ou de plusieurs) circuit(s),
- .**INDisponibilité de Liaison Réseau** (INDLR): cas correspondant aux défaillances du système entrainant l'indisponibilité d'une liaison réseau,
- .**PERte D'APpels** (PEDAP),

Ces trois derniers sont regroupés selon le sigle INDisponibilité PArtielle: INDPA.

II.1.2) Décomposition du logiciel

Considérant l'architecture du système, la première découpe qui vient à l'esprit est celle qui consiste à partager l'ensemble du logiciel en deux parties:
- le logiciel du processeur principal,
- celui des processeurs auxiliaires.

Pour chacun d'eux on peut effectuer deux sortes de décomposition selon qu'on s'intéresse à la répartition qui a déjà été effectuée lors du développement (ou production) du logiciel ou que l'on s'intéresse aux fonctions effectuées par ce logiciel. Ces composants sont définis comme suit:
- les composants de production: ce sont des entités de compilation et d'édition de liens qui peuvent regrouper un ou plusieurs traitements, ces composants seront appelés modules,
- les composants fonctionnels: appelés "traitements", ils sont au nombre de 37 pour le processeur principal et 30 pour l'auxiliaire, ils réalisent des fonctions bien particulières et sont non interruptibles.

Ces composants ont été regroupés en quatre grands groupes, correspondant aux grandes catégories de fonctions remplies par le système qui sont:
- téléphonie (TEL): tous les programmes assurant la commutation,
- exploitation (EXP): programmes d'exploitation: initialisation, positionnement des entités de commutation, localisation d'erreur après découverte d'une défaillance,
- défense (DEF): logiciels de surveillance du bon fonctionnement du système (test en ligne), de basculement et de reconfiguration après défaillance,
- moniteur (MON): tout le logiciel de base.

II.2) ANALYSE DES DEFAILLANCES, PRINCIPAUX RESULTATS

58 RTA et 135 OC ont été enregistrés sur une période de 3 ans. On présente successivement l'étude des défaillances par analyse des RTA, l'étude des corrections par l'intermédiaire des OC puis une analyse combinée des RTA et des OC.

II.2.1) Analyse des défaillances (RTA)

La figure 3 donne les proportions des conditions d'apparition et des conséquences des anomalies.

Conditions d'apparition

On peut remarquer que 44% des fautes se sont manifestées suivant un scénario correspondant au cas C, mais il faut rappeler que ce cas regroupe les fautes apparues suite à défaillance matérielle ou sur coïncidence entre défaillance matérielle et commande opérateur ou basculement inefficace. Ces 44% correspondent aux fautes mal définies qui sont souvent dues à des interactions entre modules.

Néanmoins la proportion d'apparition sur "fonctionnement normal" (cas B) reste non négligeable, on peut donc en déduire que les conditions d'apparition sont assez bien réparties entre les trois catégories considérées.

Conditions d'apparition

Cas A	31 %
Cas B	25 %
Cas C	44 %

Conséquences

INDGE	13 %
PERLO	30 %
INDLR	6 %
INDCI	15 %
PEDAP	4 %
ATEXP	33 %

Figure 3: Conditions d'apparition, conséquences des anomalies

Conséquences

On remarque que 13% des défaillances entrainent une indisponibilité générale, ces fautes ont en général été introduites au niveau de la conception donc plus difficiles à résoudre. En effet ces fautes apparaissent pour la plupart à partir du moment où le système est soumis à des contraintes de charge se rapprochant des limites du système, donc en fin de période de validation.

En outre, on observe très peu de perte d'appels car ces fautes conduisent rarement à la création de RTA.

Evolution en fonction du temps

La figure 4 donne les conditions d'apparition des fautes en fonction du temps et la figure 5 donne les conséquences des fautes en fonction du temps.

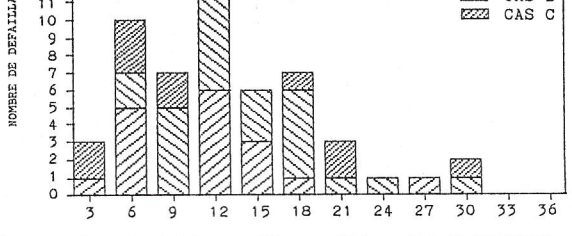

Figure 4: Conditions d'apparition des anomalies en fonction du temps (en mois)

On remarque que les fautes sur fonctionnement normal disparaissent un peu plus tôt que les autres. A l'inverse, les fautes graves demeurent assez longtemps dans le système. les fautes de programmation simples sont corrigées assez rapidemment et les fautes de spécification ou conception, qui n'apparaissent que dans des cas de surcharge ou grande sollicitation, ne sont découvertes et corrigées que plus tardivement.

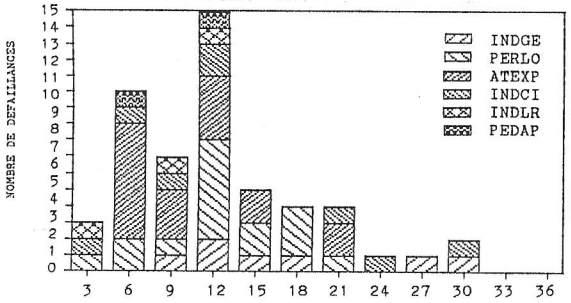

Figure 5: Conséquences des anomalies en fonction du temps (en mois)

II.2.2) Analyse des corrections (OC)

Les Groupes

On peut remarquer que les fautes sont très bien réparties entre les trois groupes: TEL, DEF et EXP (figure 6). Le groupe MON n'a subi que 15% des corrections mais il correspond au logiciel de base "système" et aux utilitaires, modules qui sont validés très tôt donc bien avant la période d'observation considérée.

La distinction entre logiciel "principal" et logiciel "auxiliaire" permet de mettre en évidence une plus grande proportion de fautes dans le groupe (EXP) sur le processeur principal et une plus grande proportion de fautes de téléphonie (TEL) sur le processeur auxiliaire. Ceci peut être expliqué par le fait que le logiciel d'exploitation du principal est plus volumineux que celui de l'auxiliaire, et le logiciel de téléphonie de l'auxiliaire est plus volumineux que celui du principal.

	Volume	Fautes
TEL	33 %	29 %
EXP	28 %	26 %
DEF	26 %	30 %
MON	13 %	15 %

Figure 6: Proportions du volume et de fautes par groupe

Les Traitements

Pour chaque groupe, on a représenté les traitements concernés par les fautes. On montre à la figure 7 à titre d'exemple le cas du groupe EXP. On remarque que dans ce groupe qui comporte 15 traitements, le traitement Tpocur contient à lui seul 45% des fautes et 3 traitements possèdent 90 % des fautes. Ceci s'explique par le fait que ces traitements sont les principaux traitements du groupe EXP et ils sont nettement plus volumineux que les autres.

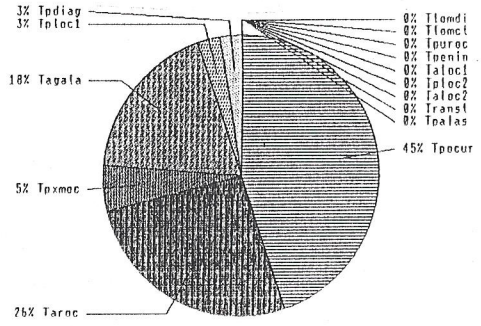

Figure 7: Traitements concernés par les fautes dans le groupe EXP

La figure 8 donne le nombre de fautes par traitement tous les groupes confondus

Environ 50 % de l'ensemble des traitements possèdent au moins une faute et seulement 20 % ont plus de trois fautes. Le logiciel a été validé progressivement et un certain nombre de traitements étaient déjà validés avant la période considérée. Les traitements d'exploitation et de défense sont plus concernés par les fautes car ils sont activés beaucoup plus rarement que les traitements du groupe TEL et les fautes mettent plus longtemps à apparaître et à être éliminées.

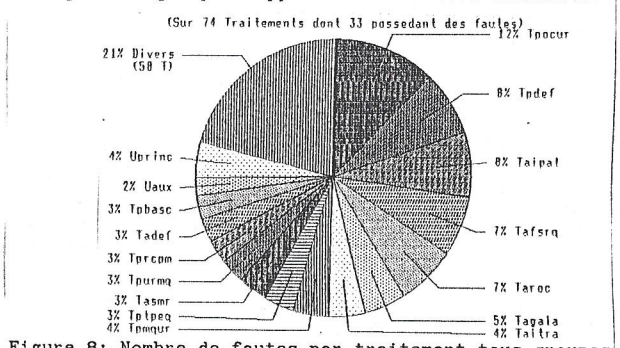

Figure 8: Nombre de fautes par traitement tous groupes confondus

Pour chaque composant qui contenait au moins trois fautes on a noté sur la figure 9 le nombre de fautes par kilo-octets cumulées sur la période de trois ans considérée. On peut remarquer que les différents taux sont compris entre 0.9 et 4.5 fautes par KO, ce qui donne simplement un rapport de 5 entre le meilleur et le plus mauvais traitement. Ceci peut laisser penser que, d'une manière générale, le nombre de défaillances découvertes par composant sur les périodes de validation et de début de vie opérationnelle est sensiblement proportionnel au volume brut du traitement. Ceci confirme les résultats parus dans ROS 82.

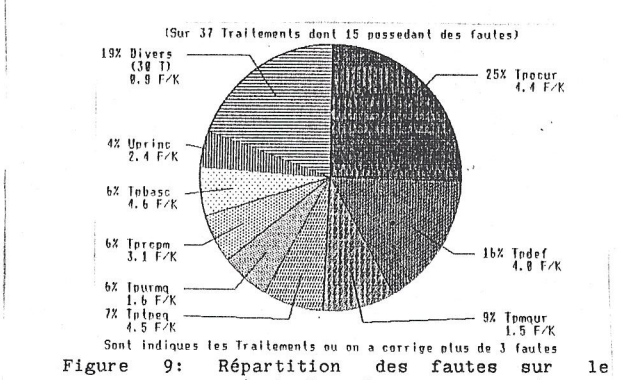

Figure 9: Répartition des fautes sur le , principal, volumes correspondants

La figure 10 montre l'évolution du nombre de corrections par groupe en fonction du temps. On constate comme prévu un léger décalage dans le temps de la validation du logiciel d'exploitation.

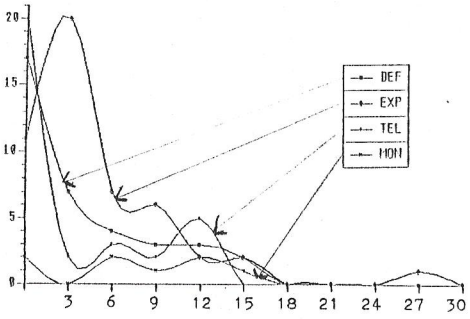

Figure 10: Nombre de corrections par groupe en fonction du temps

II.2.3) Analyse combinée des RTA et OC

Le but de cette partie est, à partir des RTA et des OC, d'établir les liens qui existent entre les conditions d'apparition des anomalies, leurs conséquences et le composant logiciel concerné. On cherche par exemple à déterminer les composants et les types de sollicitation entrainant des défaillances graves.

.Corrélations traitements ou groupes, conditions d'apparitions: Les fautes du logiciel DEF sont toujours survenues sur cas C. La défense ne semble pas

avoir d'effet négatif en cas de bon fonctionnement. En outre les fautes dans les logiciels TEL et EXP sont apparues pour moitié sur commande opérateur et pour moitié sur fonctionnement de téléphonie (cas B).

.Corrélations conséquences des fautes avec modules et traitements: Les commandes opérateur activent des fautes dont les conséquences ne sont pas graves et ne bloquent pas le système. La mise au point des dernières commandes opérateur peut donc déborder sur le début de vie opérationnelle sans trop dégrader le fonctionnement du système.

.Corrélations conséquences des fautes-conditions d'apparition: Les fautes graves proviennent pour la plupart de défaillances matérielles ou de coïncidences pendant ou après le basculement de logique. Les composants de mise à jour et de reconfiguration après basculement doivent être validés avec grand soin avant la sortie du produit sur le marché.

II.2.4) Evolution du nombre de corrections et du nombre de systèmes en service

Le nombre cumulé de corrections en fonction du temps est représenté à la figure 12. Cette courbe ne tient pas compte du nombre de systèmes en service qui n'a cessé de croître durant la vie opérationnelle. Le taux de correction en fonction du temps ramené à un système est donné à la figure 13. Le taux de correction résiduel en vie opérationnelle est de $4*10^{-8}$/Heure*Koct.

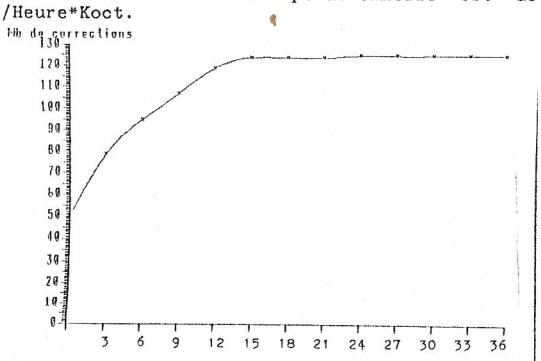

Figure 12: Nombre cumulé de corrections en fonction du temps

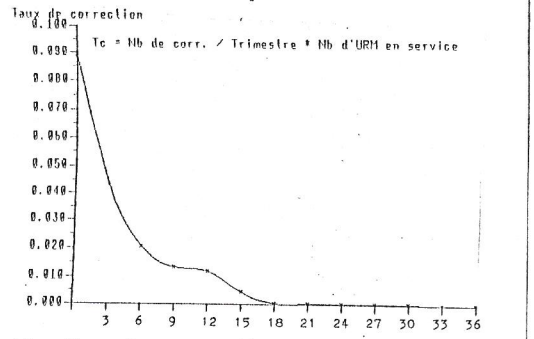

Figure 13: Taux de correction en fonction du temps par système et par trimestre

II.2.5) Conclusions

L'ensemble des résultats détaillés dans les parties précédentes nous amène à queques conclusions globales sur les différentes fautes de logiciel.

Tout d'abord on a pu remarquer que la mise en oeuvre d'un système redondant complexe ne semblait pas être trop pénalisante, bien que la mise au point nécessite un effort important. D'une manière générale les fautes sont apparues sur sollicitations importantes du système. Sollicitations pouvant provenir de l'opérateur, d'une défaillance matérielle ou d'une charge importante au niveau téléphonie. On a pu d'ailleurs constater un décalage dans le temps des déverminages des différents composants suivant leurs fréquences d'activation.

Le taux de faute logiciel semble à peu près proportionnel au volume. Sur trois ans de validation puis de vie opérationnelle on a observé de 4 à 8 fautes par kilo-octets (sur l'ensemble des systèmes en validation puis en exploitation).

Le taux de défaillance (calculé à partir des feuilles de correction) est de $4*10^{-8}$ $H^{-1}KOct^{-1}$.

Les fautes de programmation sont en général apparues en début de validation et les fautes persistantes du logiciel sont les fautes de conception ou spécification plus difficiles à corriger. Ces résultats sont en accord avec ceux de GLA 81.

Cette analyse qualitative a permis d'établir un lien entre les composants concernés et les différentes conséquences des anomalies. L'analyse quantitative effectuée dans la partie suivante consiste à tester la croissance de fiabilité puis à appliquer les modèles de croissance de fiabilité du logiciel.

III) APPLICATION DES MODELES DE CROISSANCE DE FIABILITE

III.1) DONNEES UTILISEES

Les modèles classiques de fiabilité de logiciel travaillent à partir des temps observés jusqu'à défaillance. Dans notre cas ces intervalles seront extraits des RTA et des OC. Ces intervalles seront en fait de deux types: intervalles de temps entre créations de RTA et intervalles de temps entre corrections. On obtient sur la période d'observation un ensemble complet de 58 données obtenues à partir des RTA et un ensemble de 136 données obtenues à partir des OC.

La période d'observation étant répartie entre la période de validation et la période de vie opérationnelle, et le nombre de systèmes étant croissant durant la vie opérationnelle, on ramène tous les intervalles de temps observés à un système.

III.2) TEST DE CROISSANCE

Deux tests complémentaires ont été effectués: le test de la moyenne arithmétique (Kan 86) puis le test de Laplace (Asc 78).

Pour le premier test on calcule τ_k la moyenne arithmétique des t_i observés jusqu'à la k ième donnée: Si τ_k forme une suite croissante (décroissante) on en déduit une croissance (décroissance) de fiabilité.
On a pu remarquer que dans une première phase il n'y a pas globalement croissance de fiabilité. Ceci peut s'expliquer par le fait qu'on se trouve en période de validation durant laquelle on teste constamment des nouvelles parties du système. Ensuite le système entre dans une phase de vie opérationnelle. Dans un premier temps, la croissance n'est pas évidente car surviennent quelques fautes qu'il était pratiquemment impossible d'activer en phase de simulation. Ensuite on peut alors constater une phase de vie opérationnelle stabilisée durant laquelle la croissance de fiabilité est très nette, pour arriver à un taux de défaillance très faible.

Pour le deuxième test on calcule l'indicateur de tendance de Laplace u. La croissance (décroissance) de fiabilité est caractérisée par u négatif (positif).

La figure 14 donne les résultats du test de tendance appliqué aux M premières données puis aux ensembles complets de données.

R T A

M	u
43	+1.04
48	+0.23
58	-3.19

O C

M	u
100	-4.16
120	-6.89
136	-13.1

Figure 14: Résultats du test de tendance de Laplace

Ce test permet d'affiner les résultats obtenus par le premier test. On peut remarquer que sur les ensembles complets de données, les indicateurs sont négatifs et dénotent une nette croissance globale de fiabilité.
On constate aussi une période intermédiaire où il n'y avait pas croissance et même décroissance. Ceci nous permet de situer la discontinuité due à la transition validation - vie opérationnelle.

III.3) MODELES DE CROISSANCE DE FIABILITE UTILISES

L'application de modèles de fiabilité sur un cas industriel donné apporte de nombreux renseignements sur la maîtrise du développement du système. Elle permet de:

- rechercher un modèle adéquat pour les données considérées, en effet un modèle n'est ni bon, ni mauvais dans l'absolu, il peut être bon pour un type de données (Kei 83),
- évaluer la fiabilité du logiciel au cours du temps,

on se donne une cible de fiabilité et on aide à la gestion des délais de validation,
- prévoir la fiabilité en vie opérationnelle, en intégrant dans les calculs de fiabilité du système la part due au logiciel.

On applique trois modèles aux données collectées. On fait ensuite des remarques sur l'application de ces modèles et le choix de l'un d'entre eux, remarques basées sur l'observation des résultats et intéressant directement l'utilisateur.

III.3.1) Les modèles

On considère trois modèles correspondant aux grands types que l'on rencontre dans la littérature:
- un modèle poissonnien par morceaux, la séquence des taux étant déterministe: le modèle de MUSA (MU) (Mus 75).
- un modèle exponentiel par palier où la séquence des taux de défaillance est une variable aléatoire: le modèle de LITTLEWOOD-VERRAL (LV) (Lit 73),
- un modèle basé sur un Processus de Poisson non Homogène (NHPP), il correspond à un lissage de courbe de taux de défaillance, constants par palier, obtenues par les deux autres modèles: le modèle HyperExponentiel (HE) (Kan 85).

III.3.2) Utilisation des modèles, capacité prédictive

L'application de ces modèles consiste dans un premier temps à tester l'adéquation de ces modèles aux données collectées. Chacun d'entre eux possède plusieurs paramètres estimés à l'aide du critère du maximum de vraisemblance. Pour tester l'adéquation des modèles on considère l'ensemble des MT données $t_1, t_2, .. , t_{MT}$, observées. On considère ensuite un sous ensemble de $M_\emptyset$ données tel que $1 < M_\emptyset < MT$, on estime alors, les paramètres des modèles à partir des $M_\emptyset$ premières données. On en déduit alors une estimation du $M_\emptyset+1$ ième MTTF: $\hat{t}_{M\emptyset+1}$ et on compare avec la valeur observée: $t_{M\emptyset+1}$. On refait la même séquence pour M valant de $M_\emptyset+1$ à MT, on estime à chaque pas le MTTF suivant soit: $\hat{t}_{M+1}$. On évalue la qualité de l'estimation par un critère de validation tel que le critère des résidus ou le critère de Kolmogorov-Smirnov. Le critère des résidus calcule l'écart relatif moyen entre les valeurs observées et les valeurs estimées:

$$RMR = \sum_{1}^{MT}(\hat{t}_i - t_i) \; / \; \sum_{1}^{MT} t_i .$$

Le critère de K-S s'applique aux fonctions de distribution des variables aléatoires T_i (COX 66).

III.3.3) Application des modèles, Résultats

Sur les deux ensembles de données correspondant aux défaillances et aux corrections, on applique successivement les trois modèles présentés plus haut.

Les défaillances

Sur la figure 15 on a rassemblé les résultats des trois modèles pour les données issues des RTA.

La première colonne indique le numéro de l'anomalie, la seconde l'intervalle entre défaillances correspondant observé, les troisième, quatrième, cinquième colonnes donnent respectivement les résultats des estimations du MTTF par les modèles MU, LV, HE. On note ensuite les valeurs des Résidus Moyens Relatifs et des Distances de KS pour les modèles LV et HE. Pour le modèle MU, seul un RMR partiel a pu être calculé sur les données où le modèle était applicable. Ce Résidu ne pourra pas être comparé aux autres. En effet le modèle MU n'est applicable qu'au cas où les observations vérifient une condition (LIT 81) se rapprochant du test de croissance de Laplace, condition qui n'est pas toujours remplie dans notre cas.

On observe ici une longue période durant laquelle il n'y a pratiquemment pas croissance de fiabilité; A l'inverse du modèle de MU, les modèles de LV et HE montrent leur faculté à suivre une constance ou une décroissance de fiabilité. On peut remarquer que le modèle de LV donne des résultats (RMR et DKS) plus faibles que le modèle HE et donc s'adapte mieux aux données étudiées. En contre partie le modèle de LV suit "trop" bien les données et ne réalise pas le lissage nécessaire, que donne l'autre modèle. Ce lissage permet d'avoir une valeur plus stable du MTTF estimé et de ne pas tenir compte des fluctuations (microscopiques) au cours de la période, fluctuations dues pour la plupart à des périodes de ralentissements ponctuels de l'activité de mise au point. Le modèle HE éprouve de grandes difficultés à suivre une forte croissance survenant brusquement.

'AAS1333.ASF2.MK.RES'

**** MODELES DE MUSA LITTLEWOOD-VERRAL HYPEREXPONENTIEL *****

'SYSTEME 2'

MU: N_0 = 25 T_0 = 500

LV: α = 1.4 β_1 = 2600 β_2 = 120

HE: w = 0.57 Z_1 = 2.4E -3 Z_2 = 1.4E-4

MT = 58 L_0 = 13 FE = 60

M	t_M	MTTF(M) MU	MTTF(M) LV	MTTF(M) HE
14	724.18	********	9767.47	6067.53
15	9603.34	********	7793.89	5657.81
16	739.84	********	9122.64	5939.70
17	740.96	********	7455.37	5592.89
18	9052.57	********	8898.41	5289.31
19	17137.82	********	7312.60	5510.43
20	21848.32	7354.32	9835.09	6157.33
21	4073.95	11566.99	12105.22	6983.65
22	815.91	9946.09	11460.43	6837.44
23	4935.73	8246.75	10699.98	6550.75
24	33783.77	8341.63	10149.19	6477.35
25	867.37	14715.06	13565.34	7667.06
26	16883.87	13038.93	13026.66	7380.08
27	889.74	14090.78	13694.50	7761.98
28	2679.30	11818.92	13209.32	7497.33
29	894.22	10322.62	12519.06	7318.48
30	2692.72	8930.64	11721.46	7088.46
31	898.69	8050.36	10762.23	6937.43
32	24122.26	7226.80	9304.20	6735.65
33	928.90	9742.44	12396.88	7297.69
34	5613.66	8655.83	11405.31	7098.27
35	1875.69	8298.25	10938.12	7053.03
36	4717.20	7561.95	9860.38	6900.61
37	4745.17	7259.55	9499.38	6837.76
38	2857.17	6991.67	9151.96	6779.68
39	953.51	********	8477.50	6674.15
40	26560.28	********	7359.80	6523.40
41	4946.54	********	10208.11	7041.19
42	990.43	7870.14	10206.73	6984.64
43	1985.33	7202.04	8966.05	6838.71
44	993.78	********	8237.87	6723.86
45	26594.58	********	7166.16	6589.89
46	23044.56	8303.83	9835.56	7047.85
47	12730.85	10027.36	11563.32	7400.60
48	5332.49	10415.78	11929.49	7516.81
49	7520.30	10054.85	11607.68	7469.57
50	21934.06	10159.09	11484.75	7469.57
51	3300.18	11410.98	12420.81	7767.30
52	1101.18	10684.96	12032.77	7677.18
53	108167.45	11426.19	11547.07	7548.01
54	25732.42	26662.63	16668.80	9482.96
55	16118.65	28222.96	17536.17	9787.65
56	74578.37	33967.49	17634.61	9906.81
57	3914.34	40851.40	20766.66	11028.70
58	111542.16	52754.25	20430.16	10899.27
RMR:		*1.6137113E-01*	2.1786283E-01	4.9813003E-01
DKS:			1.4808694E-01	2.0036524E-01

Figure 15: Application des modèles aux 58 données issues des RTA

Les corrections

Les résultats obtenus sont sensiblement similaires à ceux obtenus à partir des défaillances. Les modèles de LV et HE restent pessimistes, le modèle MU étant très optimiste. L'ordre de grandeur de $N_\emptyset$ est relativement bon mais le modèle ne parvient pas à suivre une grande croissance de fiabilité (à partir de la transition validation - vie opérationnelle), prévoyant des MTTF infinis et donc des taux de défaillances nuls. Le modèle HE reste très pessimiste éprouvant de très grandes difficultés à suivre une grande croissance de fiabilité vers la fin après une période de faible croissance.

Elimination progressive des premières données

Cette forte croissance correspond sensiblement au passage de la phase de validation à la vie opérationnelle. La discontinuité que l'on rencontre pose des problèmes d'applicabilité des modèles. Aussi, par la suite, on a réalisé une élimination progressive des premières données. Ce qui nous a permis de diminuer progressivement l'influence restrictive et pénalisante des données de début de validation.

III.3.4) Conclusions, problèmes rencontrés

L'application d'un modèle quel qu'il soit n'est pas immédiate. En effet trouver les paramètres d'entrée initialisant le programme de maximisation s'est avéré long et difficile et parfois même impossible. Les trois modèles sont en général pessimistes (le résidu est positif) et les résultats sont du même ordre de grandeur. Ceci nous amène à conclure que les modèles sont sensiblement équivalents. Ainsi le choix de modèle sera basé sur d'autres critères:
- l'évolution du taux de défaillance à l'infini,
- la faculté de lissage,
- la facilité d'utilisation.

Par la suite on choisit d'utiliser le modèle HE pour faire les évaluations. Ce modèle assez pratique d'utilisation, lisse très bien les données, mais reste très pessimiste; défaut qui a pu être contourné en éliminant progressivement des données de début de période d'observation. De plus on peut remarquer que ce modèle tend vers un modèle exponentiel après une certaine période de temps alors que les deux autres supposent que le MTTF devient infini (taux de défaillance nul). On peut alors envisager d'utiliser le modèle HE pour la phase de vie opérationnelle.

IV) BASES POUR UN MODELE DE FIABILITE EN VIE OPERATIONNELLE

Dans un premier temps le modèle structurel de fiabilité de logiciel (Lap 84) est rappelé. La suite du paragraphe est consacrée à la vérification de l'adéquation de ce modèle aux données collectées.

IV.1) LE MODELE STRUCTUREL

On considère un logiciel constitué de n composants. Soit π_i la probabilité d'occupation du composant i en régime stabilisé et q_i le taux de défaillance de ce même composant. Le taux de défaillance équivalent du logiciel est (Lap 84, Lit 79):

$$q_{eq} = \sum_{i=1}^{n} \pi_i q_i , \qquad (1)$$

$\pi_i q_i$ est le taux de défaillance équivalent apparent du composant i. En tenant compte des différentes conséquences des défaillances, on a:

$$q_i = \sum_{j=1}^{Ncons} q_{ij} , \qquad (2)$$

avec q_{ij} le taux de défaillance selon la conséquence j du composant i et avec Ncons le nombre de conséquences définies pour le système concerné.

IV.2) DONNEES UTILISEES

On décompose l'ensemble complet d'intervalles de temps entre défaillances issues des RTA en quatre sous-ensembles selon les conséquences: INDGE, PARLO, INDPA et ATEXP. Ce choix a été effectué afin de réaliser un compromis entre nombre de composants, volume des composants et informations disponibles sur le logiciel.

A partir de l'ensemble complet d'intervalles entre corrections on effectue une deuxième décomposition en quatre ensembles correspondant aux quatre grands groupes de composants: TEL, EXP, DEF, MON.

IV.3) OBTENTION DES PARAMETRES

Dans la partie trois on a vu l'application du modèle aux données globales concernant les RTA et les OC. Le taux de défaillance stabilisé du système correspond au MTTF estimé en fin de période de collecte. On peut en effet supposer que le taux de défaillance du logiciel sera alors sensiblement constant (très faible) car il n'y aura pratiquemment plus de correction.

On applique le modèle HE sur les quatre ensembles de données selon la première décomposition (par conséquences) et sur l'ensemble complet de données issues des RTA puis selon la deuxième décomposition (par composants) et sur l'ensemble complet de données issues des OC.

Résultats

Les tableaux des figures 16 et 17 donnent par composant puis par conséquence le nombre MT de données soumises au modèle, le nombre $L_\emptyset$ de données éliminées au cours du temps, le nombre $M_\emptyset$ de données utilisées pour faire la première estimation, le dernier MTTF estimé en fin de période et enfin le résidu moyen relatif (RMR) obtenu sur l'ensemble des données.

COMPOS.	MT	$L_\emptyset$	$M_\emptyset$	MTTF(MT)	RMR
TEL	41	0	10	33739	0.74
DEF	38	0	10	16222	0.71
EXP	48	0	10	28868	0.65
MON	16	0	6	50015	-0.6
O.C.*	136	0	16	9636	0.75

* Ensemble des données relatives aux corrections.

Figure 16: Résultats par composants

CONSEQ.	MT	$L_\emptyset$	$M_\emptyset$	MTTF(MT)	RMR
INDGE	7	0	3	140527	0.37
PERLO	16	0	6	57842	0.50
ATEXP	19	0	6	43533	0.41
INDPA	18	0	6	60548	0.54
R.T.A.*	58	0	15	15840	0.75

* Ensemble des données issues des RTA.

Figure 17: Résultats par conséquences

Commentaires

Prenons pour commencer le cas des composants. On estime leur taux de défaillance apparent (tenant compte du taux d'activation) par :

$\pi_i q_i = 1/MTTF_i(MT)$,

En utilisant (1) et en effectuant le calcul on a:

$q_c = 1.46\ 10^{-4}$.

Sur l'ensemble des données l'estimation du taux de défaillance donne:

$q_\emptyset = 1/MTTF_\emptyset(MT) = 10^{-4} \approx q_c$.

Prenons maintenant le cas des conséquences:

$q_s = 6.4\ 10^{-5}$.

Sur l'ensemble des données:

$q_h = 6.3\ 10^{-5} \approx q_s$.

Ces résultats montrent que l'application des modèles de croissance de fiabilité pour le calcul des taux de défaillance unitaires des composants est cohérente avec le modèle structurel.

Les modèles classiques de fiabilité tels que le modèle HE semblent suffisants pour obtenir une estimation des taux de défaillances des différents composants et suivant les conséquences avec une précision semblable à celles que l'on rencontre dans le cas du matériel.

CONCLUSIONS

Cette étude a été menée à partir des informations concernant les défaillances d'un autocommutateur observées sur une longue période incluant validation et vie opérationnelle. L'application des modèles de fiabilité aux données collectées a mis en évidence cette transition validation - vie opérationnelle.

Les modèles de fiabilité ont été appliqués en gardant à l'esprit la notion de conséquences des anomalies et la notion de composants logiciels. On obtient donc des résultats de fiabilité par gravité de défaillance et par composant.

Le modèle structurel a été appliqué et a montré son aptitude à suivre le comportement d'un système susceptible de défaillir. Les modèles classiques de croissance de fiabilité (HE) sont suffisants pour une évaluation des taux apparents avec une précision semblable à celle du matériel.

REMERCIEMENTS

Les auteurs tiennent à remercier messieurs J.C.Laprie (LAAS), G.Fiche et F. Le Corre (Alcatel Commutation) pour leurs contributions tout au long de cette étude.

REFERENCES

Asc 78 H.Ascher ,H.Feingold: "Application of Laplace's test to repairable system reliability", 1er Colloque international sur la fiabilité et la maintenabilité, Paris, 19-23 Juin 1978.

Cox 66 D.R.Cox, P.A.W.Lewis : The Stasistical Analysis of Series of Events, London: Chapman & Hall, 1966.

Gla 81 R.L.Glass: "Persistent Software Errors", IEEE Trans. on Soft. Eng., vol SE-7, n°2, Mars 1981.

Kan 85 K.Kanoun, J.C.Laprie : "Modeling Software Reliability and Availability from development-validation up to operation", Rapport de recherche LAAS n° 85-042, Mars 1985, révision Août 1985.

Kan 86 K.Kanoun: "Validation de Modèles Stochastiques, Application aux Modèles de Croissance de Fiabilité du Logiciel", Rap. de Rech. LAAS n° 86-002, Jan 86.

Kei 83 P.A.Keiller, B.Littlewood, D.R.Miller, A.Sofer: "Comparison of Software Reliability Predictions", Proc. 13ème Int. Symp. "Fault-Tolerant Comp. (FTCS-13)", Milan, Italie, Juin 28-30 1983, pp. 128-134.

Lap 84 J.C.Laprie : "Dependability Evaluation of Software Systems in Operation", IEEE Trans. on Soft. Eng. Vol. SE-10, No 6, Nov. 1984, p. 701-714.

Lap 85 J.C.Laprie : "Dependable Computing and Fault-Tolerance: Concepts and terminologie". Proc. 15ème Int. Symp. Fault-Tolerant Computing, Ann Arbor, Michigan, Juin 1985, pp. 2-11.

Lit 73 B.Littlewood, J.L.Verral: "A Bayesian Reliability growth model for computer Software" J. Royal Stat. Soc., C(App. stat.), 22, 1973, pp. 332-336.

Lit 79 B.Littlewood: "Software Reliability Model for Modular Program Structure" IEEE Trans. on Reliability, vol 28, n°3 Août 1979.

Lit 81 B.Littlewood, J.L.Verral: "Likelihood Function of a Debugging Model for Computer Software Reliability", IEEE Trans. on Reliability, vol. R-30, N°2, juin 1981.

Mus 75 J.D.Musa : "A theory of Software Reliability and its application", IEEE Trans. on Software Engineering, Vol. SE-1, Sept. 1975, pp. 312-327.

ROS 82 D.J.Rossetti, R.K.Iyer: "Software-related failures on the IBM 3081: A relationship with System utilization, CRC Tech. rep. n°82.8 juin 1982.

Sab 85 T.Sabourin, G.Fiche, F.Lecorre, K.Kanoun : "Evaluation de la fiabilité et de la disponibilité d'autocommutateurs téléphoniques, Etude des anomalies" Rap. de rech. LAAS.85.099-ATD.85.331, Mai 85.