SURF-2 1 : OUTIL D'ÉVALUATION DE LA SÛRETÉ DE FONCTIONNEMENT PAR CHAÎNES DE MARKOV ET RÉSEAUX DE PETRI STOCHASTIQUES

S Metge, Martine Aguéra, Jean Arlat, Serge Bachmann, C. Bourdeau, J.-E Doucet, Karama Kanoun, J.-C Laprie, J Moreira, David Powell, et al.

\

ACTES DU 9ÈME COLLOQUE INTERNATIONAL DE FIABILITÉ ET DE MAINTENABILITÉ - LA BAULE 30 MAI - 3 JUIN 1994

SURF-2[1◊] : OUTIL D'ÉVALUATION DE LA SÛRETÉ DE FONCTIONNEMENT PAR CHAÎNES DE MARKOV ET RÉSEAUX DE PETRI STOCHASTIQUES

S. Metge M. Aguéra J. Arlat S. Bachmann C. Bourdeau J.-E. Doucet K. Kanoun J.-C. Laprie J. Moreira de SouzaD. PowellP. Spiesser

LAAS-CNRS, 7 Avenue du Colonel Roche, 31400 Toulouse - France

RÉSUMÉ *- SURF-2 est un outil d'évaluation de la sûreté de fonctionnement des systèmes matériels et logiciels grâce à la construction rigoureuse, la validation et la résolution numérique de modèles markoviens. Il a été spécialement conçu pour une approche conceptuelle des systèmes basée sur l'évaluation comparative de plusieurs solutions d'architectures vis-à-vis de leur sûreté de fonctionnement. Dans SURF-2, le comportement d'un système peut être modélisé soit par une chaîne de Markov, soit au moyen d'un réseau de Petri Stochastique Généralisé (RdPsG). La superposition d'une structure dite de récompense au modèle de comportement étudié donne la possibilité d'évaluer des mesures combinées de sûreté de fonctionnement performance ou coût.*

INTRODUCTION

L'évaluation de la sûreté de fonctionnement d'un système, dès la phase de conception, permet d'éviter des modifications tardives et donc coûteuses. De plus, l'évaluation de la sûreté de fonctionnement d'un système au cours de sa conception permet d'effectuer de manière rationnelle, les inévitables choix entre diverses solutions d'architectures.

Au cours de ces vingt dernières années, un grand nombre d'outils logiciels ont été développés dans le but d'aider aux tâches de modélisation et d'évaluation des systèmes (voir par exemple [Arlat 88] ou [Johnson 88] pour une vue d'ensemble des principaux outils). Dans ce contexte, les principales motivations pour développer un nouvel outil ont été de fournir (i) un environnement de travail modulaire, (ii) les moyens d'effectuer des vérifications structurelles en cours d'élaboration du modèle et (iii) un ensemble d'outils interactifs permettant de modéliser des architectures concurrentes dans le but d'une évaluation comparative.

Ce papier est organisé de la manière suivante. Le premier paragraphe donne un aperçu général de l'approche utilisée dans SURF-2 pour évaluer la sûreté de fonctionnement de systèmes. Le deuxième paragraphe décrit les principaux objets de la base de données gérée par SURF-2. Les différents outils qui constituent "la boîte à outils" SURF-2 et qui agissent sur ces objets sont détaillés dans le troisième paragraphe. Le dernier paragraphe présente la configuration matérielle et l'environnement d'utilisation de SURF-2 ainsi que les principales performances de l'outil.

[1◊] SURF-2 a été conçu et développé par le LAAS-CNRS avec le concours de CEP SYSTEMES et a bénéficié du soutien de l'ANVAR. Il est le fruit d'un travail collectif entrepris au LAAS-CNRS, auquel Christian Béounes, Chercheur LAAS-INRIA, a largement participé, en particulier au cours de la phase de validation. Sa contribution sur l'utilisation des réseaux de Petri dans SURF-2 a été essentielle. Il nous a malheureusement quitté le 23 avril 1993, à la suite d'une longue et pénible maladie, mais reste toujours vivant dans notre mémoire.

\

1. PRÉSENTATION GÉNÉRALE

L'évaluation quantitative de la sûreté de fonctionnement d'un système peut être décomposée en deux grandes étapes :

a) la construction d'un modèle décrivant le comportement du système étudié à partir des processus stochastiques élémentaires représentant le comportement des composants du système et leurs interactions ;

b) le traitement mathématique du modèle dans le but d'obtenir les expressions analytiques ou les valeurs numériques des mesures de sûreté de fonctionnement du système.

Concernant l'étape (a), le comportement du système est décrit dans SURF-2, (i) à l'aide de réseaux de Petri stochastiques généralisés, notés par la suite RdPsG, (se référer par exemple à [Florin 85] pour une description des RdPsG) dont le graphe des marquages permet la transformation du réseau en une chaîne de Markov équivalente à temps continu, ou (ii) directement au moyen d'une chaîne de Markov. En outre, dans le cas d'un RdPsG, il est possible de calculer des invariants de marquages et de tirs de transitions, donnant ainsi la possibilité à l'utilisateur de vérifier les propriétés structurelles du modèle et d'accroître de ce fait sa confiance dans ce modèle.

L'étape (b) décrite ci-dessus, est entièrement automatique. Le traitement de la chaîne de Markov numérisée permet d'obtenir les mesures de sûreté de fonctionnement souhaitées. Le passage d'un RdPsG à une chaîne de Markov à temps continu se fait sur la base des marquages qui sensibilisent les transitions temporisées.

La figure 1 illustre les différentes étapes intermédiaires de transformation d'un réseau de Petri stochastique généralisé en une chaîne de Markov symbolique équivalente.

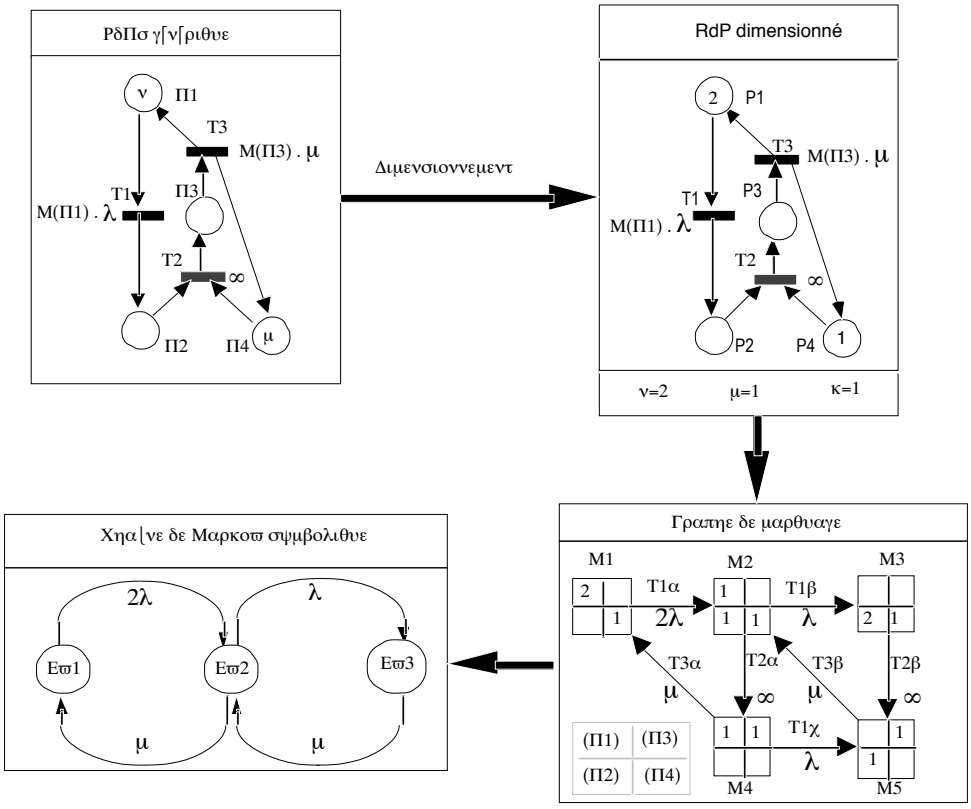


figure 1 : Transformation d'un RdPsG en une chaîne de Markov.

SURF-2 peut être assimilé à une "boîte à outils" où chaque outil peut s'exécuter indépendamment des autres et opère sur les différents objets de la base de données comme le montre la figure 2. Les deux paragraphes suivants décrivent en détail les objets regroupés dans la base de données et les outils qui les manipulent.

\

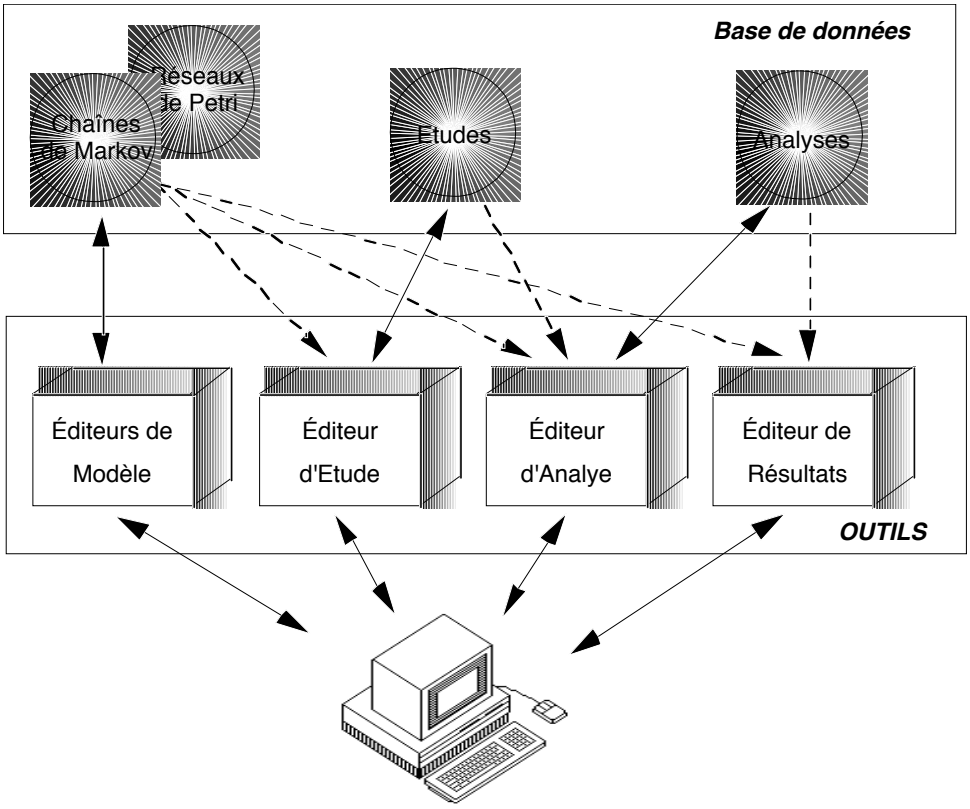


figure 2 : Structure de SURF-2 vue par l'utilisateur.

2. LES OBJETS

Les quatre principaux types d'objets créés par l'utilisateur et manipulés par les outils de SURF-2 sont les *Réseaux de Petri Stochastiques Généralisés*, les *Chaînes de Markov*, les *Études* et les *Analyses*.

Les ***Réseaux de Petri Stochastiques Généralisés*** et les ***Chaînes de Markov*** sont des *modèles* qui représentent, sous la forme d'un graphe, le comportement d'un système, tel qu'il est perçu par l'utilisateur. Ces objets contiennent également la spécification du vecteur des probabilités initiales de la chaîne de Markov (ou de la chaîne de Markov sous-jacente dans le cas d'une description du système représenté par un RDPSG) et une ou plusieurs *partitions*. Une partition consiste à spécifier de manière explicite comment la sûreté de fonctionnement du système doit être mesurée. Une partition est constituée d'une classe d'états de défaillance, dite "*improper*", et d'une classe d'états redoutés ou dangereux, dite "*catastrophic*". Ces deux classes d'états permettent de définir la plupart des mesures de sûreté de fonctionnement, la classe "*catastrophic*" étant utilisée pour l'évaluation de mesures de sécurité. Pour un même modèle, il est possible de définir plusieurs partitions afin d'étudier différents cas de fournitures de services inappropriés pour lesquels on souhaite évaluer les mesures de sûreté de fonctionnement correspondantes.

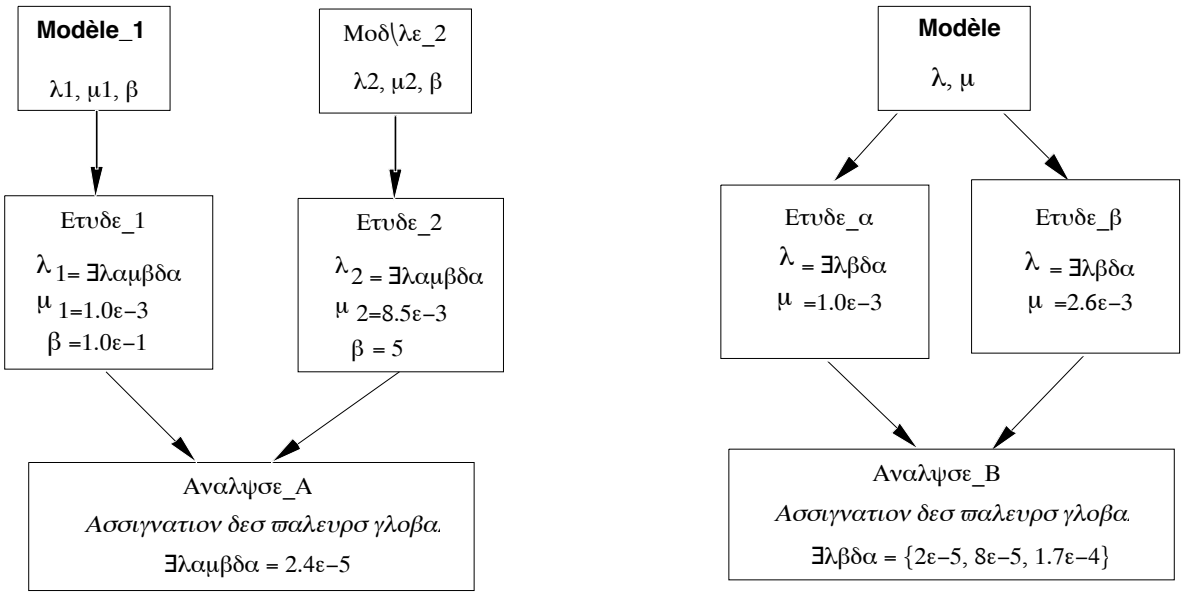
Les paramètres du modèle peuvent être des expressions numériques ou symboliques. Un paramètre symbolique est une variable locale au modèle dont la visibilité est limitée au modèle dans lequel elle a été définie. L'utilisation combinée de paramètres symboliques et la définition de plusieurs partitions pour un même modèle permet de construire des modèles génériques que l'on peut stocker dans la base de données SURF-2 en vue de les réutiliser dans d'autres modélisations.

Une ***Étude*** correspond à l'assignation d'une valeur à chaque paramètre d'un modèle. La valeur qui est assignée à un paramètre donné peut être une valeur numérique ou une ***valeur globale*** ; dans ce dernier cas, elle peut être alors commune à plusieurs paramètres de modèles différents. Cette notion de valeur globale est très similaire à la notion de variable externe, utilisée dans de nombreux langages de programmation. La généricité des modèles et la possibilité de définir plusieurs études pour un modèle donné offrent une grande souplesse de modélisation et donnent la possibilité d'effectuer des études de sensibilité d'une mesure à certains paramètres du modèle étudié.

\

La modélisation de plusieurs solutions d'architectures dans un but d'évaluation comparative, nécessite l'évaluation simultanée d'une mesure de sûreté de fonctionnement commune aux diverses solutions considérées ou aux différentes configurations d'un même système. C'est le rôle de l'***Analyse*** qui permet de spécifier la mesure à calculer, de sélectionner un ensemble d'études intervenant dans le calcul, et pour chaque étude, une partition du modèle associé à l'étude en fonction de la mesure sélectionnée. L'assignation d'une valeur (ou d'un ensemble de valeurs numériques, dans le cas d'études de sensibilité) à chaque valeur globale est réalisée au niveau de l'analyse.

La figure 3 montre deux exemples très simples d'utilisation de valeurs globales pour réaliser des études de sensibilité et des études comparatives d'architectures.



(a) Comparaison d'architectures **(b) Sensibilité à un paramètre**

figure 3 : Illustration de l'utilisation de valeurs globales.

Dans la figure 3-a deux modèles correspondant à des solutions d'architectures systèmes différentes sont comparés ; dans chacun d'eux, trois paramètres symboliques ont été définis. Le même identificateur β est utilisé pour désigner deux paramètres différents définis dans des modèles distincts. Chacun de ces deux paramètres est assigné d'une valeur numérique propre à chaque étude. A l'inverse, la valeur globale $*lambda* a été assignée à deux paramètres distincts λ_1 et λ_2 définis respectivement dans *Modèle_1* et *Modèle_2*. Chacun de ces paramètres partage donc la même valeur numérique définie au niveau de l'analyse.

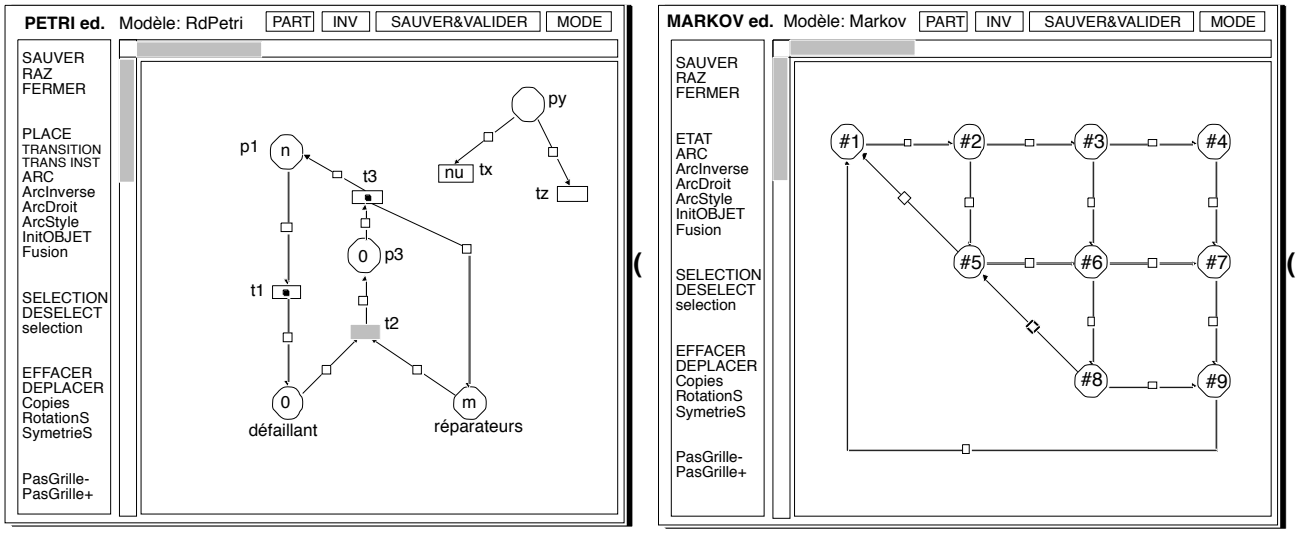
Dans la figure 3-b, deux études distinctes sont rattachées à un modèle unique. Le paramètre μ est initialisé avec une valeur numérique différente dans chacune des deux études, tandis qu'on a attribué au paramètre λ une valeur globale *$lbda*. Dans l'analyse appelée *Analyse_B*, on assigne à cette valeur globale un jeu de valeurs numériques. Dans ce cas, le calcul de la mesure effectué pour les deux études *Etude_a* et *Etude_b* fera intervenir six numérisations différentes du modèle ; la première combinaison de valeurs étant *{λ=2e-5 ; μ=1.0e-3}*. On remarquera, dans ce deuxième exemple, que l'utilisateur aurait pu aussi utiliser une valeur globale pour l'assignation du taux de réparation μ au lieu de créer deux études distinctes.

Les autres principaux objets de la base de données gérés par le superviseur de l'application sont les ***Résultats intermédiaires*** et les ***Résultats de calcul*** d'une mesure. Les *Résultats intermédiaires* sont issus du traitement d'un réseau de Petri entièrement numérisé (et dimensionné). Ils s'agit du graphe des marquage, de la chaîne de Markov équivalente sous forme textuelle, de la liste des invariants de marquage et de tir, et des conflits éventuels pour les sélecteurs aléatoires dans le cas de marquages transitoires qui sensibilisent plusieurs transitions instantanées.Les *Résultats de calcul*, pour une mesure donnée, se présentent sous forme numérique ou graphique. Tous les résultats fournis par SURF-2 peuvent être visualisés ou imprimés au format PostScript.

\

3. LES OUTILS

SURF-2 est une application multitâches : un ou plusieurs processus (deux exécutions d'un outil constituent deux processus différents), pour un même outil, ou d'outils différents, peuvent s'exécuter en parallèle et être donc confrontés à des problèmes de conflits d'accès à des données partagées. Pour éviter de telles situations conflictuelles entre outils concurrents, un contrôle de concomitance est effectué en tâche de fond par le superviseur de l'application. Certains de ces outils sont en relation directe avec les utilisateurs ; il s'agit des ***Éditeurs de modèle***. Leurs interfaces graphiques très conviviales offrent un mode de dialogue interactif. Outre les fonctions de base classiques permettant à l'utilisateur de créer, déplacer, copier ou détruire les éléments du graphe, les Éditeurs de modèles possèdent de véritables compilateurs intégrés pour l'analyse lexicale et syntaxique des expressions du modèle et des partitions. De plus, les Éditeurs de modèle vérifient en permanence que l'utilisateur n'outrepasse pas les règles prédéfinies de construction du graphe (recouvrement d'objets, fusions d'états ou de places interdites,…). Les figures 4-a et 4-b représentent respectivement l'Éditeur de réseaux de Petri et l'Éditeur de chaînes de Markov. Dans l'Éditeur de réseaux de Petri, les transitions temporisées sont représentées par des boîtes blanches tandis que les transitions instantanées sont hachurées. Les taux de transitions sont définis soit par une valeur numérique, soit par un paramètre ou une expression analytique (par exemple, le taux de la transition t_1 peut être égal à : "m(pl) * la", où "m(pl)" correspond au marquage de la place pl). Les petits carrés noirs à l'intérieur de certaines transitions temporisées indiquent que l'expression associée à la transition est trop longue pour être inscrite dans le rectangle matérialisant la transition ; dans ce cas, une simple action de la souris permet de visialiser l'expression dans une autre fenêtre.



a) Éditeur de réseaux de Petri **b) Éditeur de chaînes de Markov**

figure 4 : Éditeurs de modèles.

Un modèle provisoirement inachevé peut être enregistré dans la base de données ; par contre, il n'est pas possible d'utiliser un modèle dans un calcul dont la syntaxe n'aurait pas été préalablement validée par le compilateur. C'est le cas du réseau de Petri de la figure 4-a pour lequel aucun paramètre (ou expression) n'a été assigné à la transition t_Z.

L'***Editeur d'Étude*** a pour fonction de créer ou mettre à jour une table de valeurs associée aux paramètres du modèle qui lui est rattaché. L'utilisateur peut, à l'aide de cet éditeur, visualiser ou modifier la valeur des paramètres d'un modèle donné.

L'***Editeur d'Analyse*** permet de sélectionner un ensemble d'études, de choisir une mesure de sûreté de fonctionnement (de performance ou de coût) parmi celles proposées dans la table 1, et pour chaque étude, une partition en accord avec le type de mesure à calculer. Les mesures dites *économiques* peuvent être facilement définies au moyen d'une structure de récompense déterminée par le vecteur des taux de rendement proportionnels au temps passé dans les états du graphe, et par la matrice des bonifications acquises à chaque transition entre les états du graphe.

\

Dans le cas d'une analyse économique, on utilise le modèle de dépréciation de l'argent exponentiel décrit dans [Howard 71].

table 1: Les mesures offertes dans SURF-2.

MESURES TRANSITOIRES	*MESURES ASYMPTOTIQUES*	*VALEURS MOYENNES*
Fiabilité	*Disponibilité*	*MTFF*
Sécurité	*Indisponibilité*	*MTTF*
Maintenabilité	*Revenu*	*MUT*
Disponibilité	*Récompense*	*MDT*
Indisponibilité		*MTBF*
Récompense		*AC*
		Récompense

Les résultats numériques obtenus après traitement de l'ensemble des études pour une analyse donnée sont enregistrés dans la base de données de SURF-2. Ces résultats sont visualisés à l'aide de l'***Éditeur de résultats de calcul***. Cet outil fournit à l'utilisateur un moyen à la fois très puissant et très souple pour exploiter les résultats d'un calcul. Il offre en particulier, la possibilité de sélectionner une partie des résultats, de visualiser un faisceau de courbes dans une même fenêtre, et d'effectuer des zooms sur une portion du graphique.

De part de sa conception originale de "boîte à outils", SURF-2 offre une très grand souplesse d'utilisation donnant la possibilité d'intégrer facilement de nouveaux outils. Un guide d'utilisation [Béounes 92] documente en détail les fonctionnalités et les possibilités de chaque outil par le biais de nombreux exemples.

4. MISE EN ŒUVRE ET PERFORMANCES

SURF-2 nécessite l'utilisation de stations de travail SUN-4 (ou Sparc) avec l'environnement graphique X Window ou OpenWindows. L'application représente 70 000 lignes de code source (60 000 lignes en langage C, 10 000 lignes en Ada) et 14 Moctets de code exécutable. Grâce à la gestion dynamique de la mémoire vive, la taille des modèles à évaluer n'est pas limitée. Au cours de la validation de SURF-2, des modèles de plusieurs milliers d'états (par exemple, 12 000 états et 85 000 arcs) ont été évalués sur une station de travail Sparc IPC de 24 Moctets de mémoire et une zone de swapping de 33 Moctets. Avec une telle configuration matérielle, la taille maximum atteinte pour une chaîne de Markov a été de 20 000 états et 155 000 arcs. Une nouvelle version de SURF-2, d'ores et déjà à l'étude, permettra en particulier la prise en compte de la croissance de fiabilité et intégrera la méthode des états fictifs [Cox 68, Laprie 75] pour prendre en compte des lois de distribution non exponentielles.

RÉFÉRENCES

[Arlat 88] J. Arlat, “Méthodes et outils de la sûreté de fonctionnement des systèmes informatiques tolérant les fautes”, TSI, vol. 7, no. 4, pp.345-357, 1988.

[Béounes 92] C. Béounes *et al.*, *Manuel d'utilisation du logicel SURF-2,* LAAS-CNRS, Déc. 1992.

[Cox 68] D. R. Cox et H. D. Miller, *The Theory of Stochastic Processes*, Methuen, Londres, 1968.

[Howard 71] R. A. Howard, *Dynamic Probabilistic Systems*, vol. II, Wiley, New-York, USA, 1971.

[Johnson 88] A. M. Johnson Jr et M. Malek, “Survey of Software Tools for Evaluating Reliability, Availability and Serviceability”, ACM Comp. Surveys, 20 (4), pp.227-269, Déc. 1988.

[Florin 85] G. Florin, et S. Natkin, “*Les réseaux de Petri Stochastiques*”, TSI. vol. 4, no 1, Fév. 1985.

[Laprie 75] J.-C. Laprie, “Prévision de la sûreté de fonctionnement et architectures de structures numériques temps réel réparables”, Doctorat d'Etat, Univ. Paul Sabatier, Toulouse, 1975.