Projet REFLECS

Systèmes informatiques distribués temps réel tolérant les fautes

Localisation : *Rocquencourt*

Mots-clés : architecture répartie, algorithme de contrôle de concurrence, parallélisme asynchrone, réseau local, réseau local radio, protocole de communication, accès multiple, ordonnancement, tolérance aux fautes, consensus réparti, temps réel, application critique, système complexe, modélisation analytique, simulation, évaluation de performances, raisonnement d'adversité, méthode de conception, dimensionnement prouvé correct, génie système.

1 Composition de l'équipe

Responsable scientifique

Gérard Le Lann, Directeur de Recherche Inria

Responsable permanent

Pascale Minet, Chargée de Recherche Inria

Secrétaire

Dominique Poulicet, AI Inria

Personnel Inria

Paul Mühlethaler, Chargé de Recherche Inria, jusqu'en septembre 1996

Ingénieurs experts

Emmanuelle Anceaume, jusqu'en octobre 1996
Nicolas Rivierre

Chercheur post-doctorant

Marco Spuri, Scuola Superiore S. Anna, Pise (Italie), programme HCM, jusqu'en septembre 1996

Chercheurs doctorants

Patrice Carrère, Université Paris 6, boursier Inria
Laurent George, Université Versailles Saint-Quentin, boursier Inria
Jean-François Hermant, Université Paris 6, boursier Inria

Collaborateurs extérieurs

Bernadette Charron-Bost, CNRS/ LIX, Palaiseau
Bruno di Gennaro, ESIGETEL

Visiteurs (contrats de coopération)

Leila AZOUZ, ENSI, Tunis
Marcelo STEMMER, Univ. Fédérale de Santa Catarina, Brésil

Stagiaires

Sonia Mettali, ENSI, Tunis

2 Présentation du projet

Les travaux du projet Reflecs concernent les systèmes informatiques complexes servant à mettre en œuvre des applications critiques, c'est-à-dire des applications dont les défaillances sont inacceptables ou catastrophiques. Ces travaux se répartissent en deux catégories, l'une méthodologique, l'autre algorithmique. Les travaux de la première catégorie portent sur les méthodes de Génie Système prouvable, pour des problèmes applicatifs de type déterministe. L'obtention des propriétés applicatives souvent exigées dorénavant par les donneurs d'ordre ou les utilisateurs impose de résoudre des problèmes algorithmiques de type "traitement distribué" (TD) ou/et "temps réel" (TR), ou/et "tolérance aux fautes" (TF). Les travaux de la deuxième catégorie ont pour but de résoudre de façon prouvable de tels problèmes.

Les propriétés applicatives recherchées sont la sûreté ("safety"), la vivacité ("liveness"), la ponctualité ("timeliness") et la confiance ("dependability"). Ces propriétés peuvent être quantifiées de diverses manières. Par exemple, bornes supérieures de temps de réponse ou gigues bornées sont des mesures de ponctualité, tandis que disponibilité ou fiabilité sont des mesures de confiance.

Le but général de nos travaux est de pouvoir élaborer une spécification modulaire de système informatique distribué (une solution) qui, de façon prouvable, satisfait une spécification de besoins applicatifs et de propriétés exigées (un problème informatique), cette dernière étant dérivée d'une description initiale éventuellement incomplète et/ou ambigüe fournie par un donneur d'ordre ou un utilisateur. Par "spécification", il faut comprendre l'expression (en langage naturel, en notations formalisées) complète et non ambigüe d'un problème, d'une solution.

Les thèmes couverts sont définis comme suit :

- Traitement Distribué (TD) : un système est dit distribué si les services qu'il fournit sont obtenus par le biais d'algorithmes explicitement conçus pour être exécutés en parallèle, de façon asynchrone, par plusieurs processeurs, sans connaissance de l'état global du système (propriétés de sûreté et vivacité). La complexité des problèmes algorithmiques va croissant, en allant du modèle (de calcul) synchrones aux modèles partiellement synchrones, puis au modèle asynchrone. Nous nous intéressons uniquement aux solutions algorithmiques déterministes.

- Temps Réel (TR) : un système est dit temps réel si ses spécifications comportent des références directes ou indirectes au temps physique et si son comportement est déterminé par des algorithmes d'ordonnancement utilisant directement ou indirectement des attributs temporels dérivés des spécifications (propriétés de ponctualité). La complexité des problèmes algorithmiques et des conditions de faisabilité va croissant, en allant du modèle (de contrainte temporelle) à échéances de terminaison au plus tard constantes au modèle à gigues de terminaison bornées et à dates de démarrage au plus tôt non constantes (il existe des modèles intermédiaires), et du modèle (événementiel) périodique au modèle arbitraire (il existe des modèles intermédiaires). Nous nous intéressons exclusivement aux solutions algorithmiques déterministes.

– Tolérance aux Fautes (TF) : un système est dit tolérant aux fautes si son comportement reste conforme à sa spécification malgré la présence d'états erronés (logiciel, matériel, données) et malgré l'occurrence de défaillances partielles d'origine interne ou dues à l'environnement (propriétés de confiance). La complexité des problèmes algorithmiques va croissant en allant du modèle (de défaillance) de type arrêt immédiat au modèle de type arbitraire ou byzantin (il existe des modèles intermédiaires). Nous nous intéressons exclusivement aux solutions algorithmiques déterministes et, le cas échéant, à l'établissement de résultats d'impossibilité.

3 Actions de recherche

3.1 Génie système prouvable et méthode TRDF

Participants : Gérard Le Lann, Pascale Minet

L'évolution des besoins applicatifs (appels d'offre, cahiers des charges) impose de plus en plus souvent de résoudre des problèmes informatiques de nature déterministe. Les propriétés exigées (sûreté, vivacité, ponctualité, confiance) le sont pour des scénarios pires cas – qu'il s'agit d'identifier et de prouver – jouables par des environnements modélisés selon des "adversaires" déterministes, par opposition aux "adversaires" stochastiques considérés en modélisation analytique (ex : théorie des réseaux de files d'attente) ou en analyse statistique (ex : simulation événementielle). L'incertitude découlant de l'impossibilité de prédire le futur de façon certaine est "encapsulée" dans des hypothèses (certains des modèles brièvement décrits au §.2). Pour des hypothèses données, on exige des preuves de propriétés certaines obtenues grâce à des conceptions reposant sur des solutions algorithmiques déterministes. Nos travaux de l'année 1996 nous ont permis de mieux comprendre ce que recouvre le génie système prouvable, ainsi que d'enrichir la méthode TRDF.

Le génie système prouvable (en informatique) sert à dérouler correctement les phases suivantes d'un projet informatique :

– capture des besoins applicatifs

  Il s'agit de la traduction d'une description non (ou partiellement) quantifiée U' (éventuellement incomplète et ambiguë, fournie par un "client") de besoins applicatifs en une spécification non (ou partiellement) quantifiée de problème informatique U "contenu" dans $U^{'}$;

– conception système

  Il s'agit de dérouler les étapes de conception permettant d'aboutir à une spécification d'un système informatique non (ou partiellement) dimensionné US qui, de façon prouvable, résout U. Pour franchir une étape, il faut satisfaire des obligations de preuves (de conception correcte) ;

– dimensionnement de système

  Il s'agit, à partir d'une quantification V' (fournie par le client) de la description U', d'élaborer la quantification correspondante V du problème U, et d'en déduire le dimensionnement VS du système-solution US. Pour obtenir VS, il faut satisfaire des obligations de preuves (de dimensionnement correct).

Ainsi, $\langle US, VS\rangle$ résout de façon prouvable le problème $\langle U, V\rangle$, capture des besoins $\langle U', V'\rangle$. Ce n'est qu'une fois que l'on dispose d'une spécification modulaire $\langle US, VS\rangle$ d'un système informatique complet que l'on peut mettre en œuvre les autres disciplines d'ingénierie aux fins d'implanter les modules en question (génie électrique, génie optique, génie logiciel, etc.).

La méthode TRDF est une méthode de génie système prouvable. En 1996, elle a été mise en œuvre sur les quatre cas réels suivants :

– avionique modulaire (contrat DRET en partenariat avec Dassault Aviation), pour la phase de dimensionnement correct de la solution US fournie en 1995 (algorithmique TRDF ORECA sur architecture distribuée),

– sûreté des centrales nucléaires (contrat IPSN), pour les phases de capture des besoins applicatifs et d'analyse de la conception d'un système informatique du commerce, aux fins de démonstration de l'adéquation ou de l'inadéquation du système considéré,

– échec du vol 501 d'Ariane 5 ; à partir du scénario de défaillance décrit dans le rapport de la commission d'enquête, il a été possible d'identifier les fautes de génie système commises lors des phases de capture, conception et dimensionnement. Ces fautes sont les causes de la défaillances du vol 501, les erreurs de génie logiciel n'étant que des conséquences de ces fautes.

– fonction trains-trappes du système atterrisseur du RAFALE (programme GENIE, contrat MENESR en partenariat avec Dassault Aviation) pour l'analyse de la tolérance aux fautes de la solution proposée.

3.2 Algorithmique TRDF

Sans algorithmique, les systèmes informatiques n'existeraient pas. Le nombre d'applications informatiques dont les spécifications contiennent des exigences combinées de Temps Réel (TR), de Traitement Distribué (TD) et de Tolérance aux Fautes (TF), d'où l'acronyme TRDF, va croissant. On peut citer des exemples récents d'appels d'offre dans des domaines aussi divers que le contrôle de trafic aérien, les marchés financiers, l'avionique ou la défense des bâtiments de surface.

Les logiciels applicatifs sont de plus en plus souvent vus comme des assemblages de composants logiciels développés indépendamment les uns des autres et qui partagent des ressources communes, telles des objets modifiables et persistants (des données) dont les états doivent vérifier des invariants globaux I. Le problème majeur posé par de telles contraintes (TD) est de prouver qu'en fonctionnement opérationnel, toutes les exécutions possibles (éventuellement concomitantes) d'un nombre quelconque de composants logiciels applicatifs, sur un nombre quelconque de processeurs, vérifient des propriétés de sérialisabilité (les invariants I sont toujours satisfaits). De plus, il faut prouver que des contraintes temporelles particulières sont satisfaites par toutes les exécutions possibles (TR), le tout en présence de défaillances de type arrêt, omission, temporel (TF).

L'obtention de ces propriétés et l'établissement des preuves correspondantes reposent sur l'analyse d'algorithmes TRDF, laquelle fournit également les conditions sous lesquelles les propriétés quantifiées sont garanties.

3.2.1 Systèmes Transactionnels Répartis Temps Réel (TR^2)

Participants : Laurent George, Pascale Minet

Un système Transactionnel Réparti Temps Réel (noté TR^2) est un système composé de clients et de serveurs, interconnectés par un système de communication. Des requêtes extérieures sont reçues par les clients et doivent être exécutées par les clients et les serveurs. Chaque requête est modélisée par une transaction composée d'actions, une action devant, par hypothèse, être exécutée complètement par un seul serveur. Une transaction est représentée par le graphe de ses actions. On se restreint ici aux graphes de type arborescent. L'ordre d'apparition des requêtes extérieures est significatif. Il détermine l'ordre d'éxécution des transactions, éventuellement concomitantes. Les contraintes temporelles sont modélisées par une échéance relative stricte de terminaison affectée à chaque transaction, dont la valeur n'est pas connue à l'avance. Chaque serveur ordonnance localement les actions en attente selon l'ordre croissant des dates d'activation par les clients. Les graphes considérés étant du type arborescent, il n'existe pas de dépendances multi-serveurs (c'est-à-dire de contraintes de synchronisation inter-serveurs). Nous avons établi les bornes supérieures des temps de réponse des transactions et les conditions de faisabilité associées. Les valeurs numériques des paramètres entrant dans l'expression de ces bornes ne doivent être fournies que lors du dimensionnement du système. La solution proposée est donc générique.

3.2.2 Analyse holistique de systèmes répartis temps réel à ordonnancement par échéance la plus proche en premier

Participants : Jean-François Hermant, Marco Spuri

La théorie holistique proposée par Tindell et Clark est une approche destinée à établir la faisabilité de systèmes répartis temps réel à priorités fixes. Son mérite apparent est de permettre l'analyse de systèmes répartis temps réel, sans que celle-ci soit trop pessimiste. Dans cette étude, nous étendons la théorie holistique à l'analyse de systèmes répartis temps réel ordonnancés selon les échéances de terminaison des tâches. Le protocole à jeton temporisé (Timed Token protocol) est considéré pour l'arbitrage des accès au réseau de la part des différents processeurs. En plus, pour obtenir une bonne utilisation des ressources, nous avons supposé que les paquets émis par les processeurs sont localement ordonnancés selon l'algorithme "échéance la plus proche en premier" (EDF). Nous avons développé une procédure pour le calcul des délais de communication en pire cas. Les résultats analytiques établis dans cette étude ont été validés sur un exemple d'application distribuée temps réel, pour laquelle les pires temps de réponse des traitements de bout-en-bout sont strictement bornés. L'exemple a confirmé l'efficacité de l'algorithme d'ordonnancement EDF global.

3.3 Algorithmes d'ordonnancement

3.3.1 Analyse de l'ordonnancement par échéance la plus proche en premier

Participant : Marco Spuri

Nous avons élaboré une technique uniforme pour analyser la faisabilité de problèmes temps réel dans le cas des systèmes centralisés à ordonnancement EDF, pour toutes les lois événementielles connues actuellement. Dans sa formulation la plus générale, l'analyse considère des tâches sporadiquement périodiques à échéances arbitraires, à gigues sur les instants d'activation, et avec partage de ressources. Les coûts induits par l'ordonnancement des tâches apériodiques (à échéances non strictes) et par une implantation basée sur une horloge sont pris en compte.

En particulier, nous avons établi une procédure pour le calcul du pire temps de réponse des tâches. Bien que ce problème ait déjà été étudié pour des systèmes à priorités fixes, nous ne connaissons aucun travail sur ce sujet lorsque l'ordonnancement EDF est utilisé. L'évaluation des pires temps de réponse est fondamentale pour l'analyse de contraintes temporelles de bout-en-bout dans les systèmes distribués.

3.3.2 Analyse des ordonnancements à priorités fixes ou dynamiques dans les cas préemptif et non préemptif

Participants : Laurent George, Nicolas Rivierre, Marco Spuri

La théorie de l'ordonnancement appliquée aux problèmes temps réel pour les systèmes centralisés a produit de nombreux résultats au cours des vingt dernières années et il peut apparaître difficile de s'y retrouver face à la pléthore de résultats existants. Nous avons cherché à réunir ces résultats pour les cas centralisé, non-oisif, préemptif/non-préemptif, à priorités fixes ou dynamiques. Pour cela, nous avons considéré des jeux de tâches dont les relations entre les échéances de terminaison et les paramètres des lois d'arrivée sont quelconques. L'optimalité des politiques d'ordonnancement, les conditions de faisabilité associées ainsi que les pires temps de réponse sont donc examinés pour le cas général. Des extensions classiques telles que la gigue sur les inter-arrivées ou la présence de ressources partagées sont aussi examinées.

Il apparaît que l'ordonnancement préemptif et l'ordonnancement non-préemptif sont très proches dans l'expression des conditions de faisabilité. De plus, l'analyse des ordonnanceurs à priorités fixes ou dynamiques peut être unifiée par l'utilisation du concept de période occupée relative à une priorité (et donc dépendante de l'ordonnanceur utilisé). En particulier, nous introduisons le concept de "deadline-d

busy period" pour l'ordonnanceur EDF qui nous paraît être un point de départ intéressant pour une comparaison avec les "level-i busy period" utilisées dans le cas d'ordonnanceur à priorités fixes.

3.3.3 Comparaison des ordonnancements préemptifs statique et dynamique

Participants : Jean-François Hermant, Laurent Leboucher (CNET), Nicolas Rivierre

Il existe deux familles principales d'algorithmes d'ordonnancement temps réel, la première s'appuyant sur des priorités fixes et la seconde sur des priorités dynamiques de type échéance. Ces familles n'ont jamais été vraiment comparées l'une à l'autre, si ce n'est en termes de mise en œuvre. Par contre, on sait peu de choses sur les rapports existant entre les complexités des conditions de faisabilité d'une part, les efficacités des algorithmes d'autre part. Notre but est de comparer ces deux familles dans le cas des systèmes centralisés à préemption, en termes d'efficacité des algorithmes et en terme de complexité des conditions de faisabilité associées. Dans un premier temps, nous introduisons un cadre général basé sur une représentation de l'ordonnancement temps réel préemptif sous forme de structure algébrique ainsi que sur une évaluation précise du nombre d'opérations élémentaires contenues dans les conditions de faisabilité. Ensuite, nous appliquons la comparaison pour différents types de trafics. Il apparaît que plus les échéances des tâches sont grandes face aux périodes et plus les tâches ont des paramètres homogènes, plus la domination théorique des algorithmes à priorités dynamiques s'estompe (resp. s'accentue). De plus, il apparaît que la complexité légèrement plus faible des conditions de faisabilité basées sur des priorités fixes ne peut être considérée comme un facteur déterminant.

Il semble donc être intéressant, en fonction du problème d'ordonnancement considéré (allant du petit système embarqué jusqu'au grand système réparti) de faire intervenir ces mesures d'efficacité et de complexité pour choisir l'algorithme d'ordonnancement le mieux adapté.

Cette action est menée conjointement avec le CNET et donne lieu à une coopération (cf. §5.1) pour étendre ces résultats préliminaires au cas des systèmes distribués.

4 Actions industrielles

4.1 Avionique Modulaire

Participants : Emmanuelle Anceaume, Laurent George, Jean-François Hermant, Gérard Le Lann, Pascale Minet, Nicolas Rivierre

Cette activité menée conjointement avec Dassault Aviation, sur financement DRET, est la poursuite des travaux démarrés en 1995.

Le problème applicatif posé par Dassault Aviation concerne l'avionique modulaire future. Nous avons appliqué la méthode TRDF afin d'obtenir une spécification du problème informatique correspondant, ainsi qu'une spécification d'un système modulaire embarqué prouvé résoudre ce problème. La solution-système spécifiée en 1995 est basée sur un modèle à objets distribués modifiables persistants, pour des tâches applicatives de type graphes orientés finis quelconques et pour tout type d'architecture distribuée sans mémoire partagée. ORECA, la solution algorithmique TRDF spécifiée, combine des ordonnanceurs non préemptifs et préemptifs, oisifs et non oisifs, ainsi que des accords de type implicite ou explicite pour la sérialisabilité, le tout en présence de défaillances de type omission et arrêt. On a également spécifié OORECA, l'outillage lié à ORECA, qui contient en particulier un oracle de faisabilité, c'est-à-dire un programme qui permet de déterminer à l'avance si un problème applicatif dimensionné est faisable avec ORECA et de dimensionner le système embarqué.

Nous avons suivi l'implémentation effectuée par Dassault Aviation de ORECA et OORECA. Ceci nous a permis de préciser certains services de ORECA et certaines notions du modèle transactionnel sous contraintes temps réel. Nous avons également fourni une version optimisée de l'oracle de faisabilité.

L'évaluation de ORECA vis-à-vis de sa facilité d'implémentation est positive. Des extensions du problème posé en 1995 sont d'ores et déjà identifiées. L'évaluation de OORECA est en cours.

4.2 Avionique Temps Réel Tolérante aux Fautes

Participants : Emmanuelle Anceaume, Gérard Le Lann, Pascale Minet

Dans le cadre du programme GENIE (MENESR, Dassault Aviation) nous avons analysé les propriétés de tolérance aux fautes du système atterrisseur du RAFALE. Cette analyse a été conduite selon la méthode TRDF. Notre travail a consisté tout d'abord à énoncer de façon non ambiguë le problème applicatif posé, lequel est caractérisé par des propriétés de fiabilité et de disponibilité et des hypothèses sous lesquelles on doit obtenir ces propriétés. Le problème informatique posé étant connu, nous avons examiné deux solutions proposées par Dassault Aviation. Une solution, en Génie Système, est une spécification d'architecture et d'algorithmes qui déterminent le comportement de l'architecture. Il a été montré qu'une des solutions n'est pas acceptable. L'étude (Génie phase 1) devrait être poursuivie afin de conclure à propos de l'autre solution.

4.3 Contrôle-commande dans une centrale nucléaire

Participants : Gérard Le Lann, Pascale Minet

Cette étude a été menée dans le cadre d'un contrat avec l'IPSN. L'IPSN a sélectionné comme application temps réel critique à examiner le contrôle-commande du système élémentaire d'eau brute secourue (SEC) d'une centrale nucléaire. Nous avons déroulé la méthode TRDF, d'une part pour capturer les besoins applicatifs (donc pour spécifier le problème informatique P à résoudre) et, d'autre part, pour analyser un système de contrôle-commande commercialisé envisagé pour l'application SEC. Cette analyse nous a permis d'identifier le problème informatique P' effectivement résolu par le système commercialisé et de démontrer que P' est plus faible que P.

5 Actions nationales et internationales

5.1 Coopération avec le CNET

Participants : Jean-François Hermant, Laurent Leboucher, Nicolas Rivierre

Une infrastructure de réseau telle que l'Internet n'offre aujourd'hui à ses utilisateurs aucune garantie de qualité de service temporelle mais seulement une fonction de transport de l'information. Le développement exponentiel de ce réseau et les besoins croissants des applications multimédia suggèrent qu'offrir des services de connexions temps réel adaptés aux technologies des infrastructures sous-jacentes (réseau, système d'exploitation, machines,...) serait un avantage considérable pour les futurs opérateurs de réseau. Cette collaboration CNET/INRIA a pour objectif de compléter les résultats du projet ACTS ReTINA en fournissant des services complémentaires permettant de garantir une qualité de service de bout en bout sur un réseau d'information. L'environnement est caractérisé par l'existence simultanée et non prévisible de plusieurs applications réparties. Dans ce cas, maîtriser les techniques de contrôle d'admission s'avère essentiel. Offrir des services de type "connexion temps réel" implique d'examiner les problèmes de réservation de ressources processeurs d'extrémités et réseau, d'ordonnancement des messages et des tâches, des tests d'admission globaux associés et du contrôle dynamique de la qualité de service fournie.

L'étude porte sur l'analyse, la comparaison et le choix d'algorithmes d'ordonnancement de tâches réparties contraintes par échéances de terminaison et l'énoncé des tests de faisabilité correspondants. Au cours de l'année 1996, la collaboration a porté ou a été initialisée sur les thèmes suivants :

- l'extension aux environnements considérés des résultats obtenus pour le cas centralisé (cf. §3.3.3),
- l'examen d'algorithmes distribués non préemptifs,
- l'établissement de conditions suffisantes pour l'ordonnancement holistique.

5.2 Participation à des comités de programme

CFIP'96, Colloque Francophone sur l'Ingénierie des Protocoles, Rabat, Maroc, octobre (P. Minet)

Real-Time Systems, Paris, janvier (P. Minet)

6 Diffusion des résultats

6.1 Actions d'enseignement

- INSTN, DEA d'Informatique, cours réseaux locaux (B. di Gennaro, P. Minet)

- Université Paris 6, DESS Téléinformatique, cours réseaux locaux temps réel (P. Minet)

- ENST Paris, 3ème année, cours réseaux locaux temps réel et tolérance aux fautes (P. Minet, N. Rivierre)

- Université Paris XII, IUT Informatique, cours architecture des machines (E. Anceaume), cours réseaux locaux (E. Anceaume)

- Université de Technologie de Troyes, 3ème année, cours réseaux locaux (B. di Gennaro)

- ISTY Versailles, 3ème année, cours ordonnancement temps réel (L. George) et cours réseaux locaux temps rél (J.F. Hermant)

- ENST, Rennes, 3ème année, cours réseaux locaux et ordonnancement temps réel (B. di Gennaro)

- ESIGETEL, 3ème année, cours réseaux locaux industriels et réseaux locaux sans fils (B. di Gennaro, N. Rivierre)

- ENSI Tunis, DEA d'informatique, cours ordonnancement temps-réel (L. George), cours réseaux locaux temps réel (J.F. Hermant)

- ESIE, Noisy le Grand, 4ème année, cours Réseaux Informatiques (B. di Gennaro)

- Scuola Superiore S. Anna, Pise (Italie), 3ème année, "Real-Time, Distributed, Fault-Tolerant Technologies (TRDF) : State-of-the-Art and System Engineering" (G. Le Lann)

6.2 Jury de thèse

- Jury d'habilitation (Ken Chen), Université Versailles-St Quentin, 25 juin 1996 (G. Le Lann)

6.3 Conférences et Colloques

- Salon Real-Time Systems, Paris, janvier, "Les nouveaux systèmes embarqués : problèmes et solutions", tutoriel (G. Le Lann)

- Colloque "Application des Méthodes Formelles au Développement des Systèmes Critiques", IMAG, Grenoble, janvier 1996, "Méthodes formelles et génie système : la méthodologie TRDF" (G. Le Lann, E. Ledinot, Dassault Aviation)

- IEEE Intl. Symposium on ECBS, Friedrichshafen (D), mars (G. Le Lann)

- INCOSE'96, 6th Intl. Conference on Systems Engineering, Boston (USA), juillet (G. Le Lann)

6.4 Activités extérieures

- Séminaire Thomson-CSF/TM, Jouy-en-Josas, 15-17 avril 1996, “Temps réel, traitement distribué et tolérance aux fautes” (G. Le Lann)

- Séminaire Digital Corp., Shrewsbury, MA (USA), 9 juillet 1996, “A Proof-based System Engineering Methodology for Designing and Dimensioning Critical Complex Computing Systems” (G. Le Lann)

- Séminaire Thomson-CSF/TM, Jouy-en-Josas, 16-18 septembre 1996, “Systèmes TRDF : état-de-l’art et méthode de génie système”’ (G. Le Lann)

- Séminaire Collège de Polytechnique, Groupe Lagardère, Bougival, 11 octobre 1996, “Les bases du génie système prouvable en Informatique” (G. Le Lann)

- Séminaires CNUCE-CNR, Pise (Italie), 22-23 octobre 1996, “Proof-based System Engineering for Computing Systems”, “An Analysis of the Ariane 5 Flight 501 Failure - Comments on the Inquiry Board Report” (G. Le Lann)

- Séminaires à l’école “Embedded Systems”, (European Educational Forum), Veldhoven (Pays-Bas), 25-29 novembre 1996, “Proof-based System Engineering as a Key Discipline for Building Provably Correct Embedded Systems”, “The Ariane 5 Flight 501 Failure as a Case Study” (G. Le Lann)

- Séminaire IRISA, Rennes, 6 décembre 1996, “Une discipline en émergence : le Génie Système. Fondations scientifiques et mise en oeuvre sur des applications temps réel distribuées et tolérantes aux fautes” (G. Le Lann)

7 Publications

Communications à des congrès, colloques, etc.

[604] J.F. HERMANT, M. SPURI, « End-to-end response times in real-time distributed systems », *in : PCDS’96, 9th ISCA/IEEE International Conference on Parallel and Distributed Computing Systems*, p. 413–417, Dijon, France, 25-27 septembre 1996.

[605] G. LE LANN, « A methodology for designing and dimensioning critical complex computing systems », *in : IEEE International Symposium on the Engineering of Computer-Based Systems*, p. 332–339, Friedrichshafen, Allemagne, 11-15 mars 1996.

Rapports de recherche et publications internes

[606] L. GEORGE, N. RIVIERRE, M. SPURI, « Preemptive and non-preemptive real-time uniprocessor scheduling », *rapport de recherche n°2966*, Inria, septembre 1996.

[607] G. LE LANN, « The Ariane 5 Flight 501 Failure - A case study in system engineering for computing systems », *rapport de recherche*, Inria, décembre 1996, rapport technique.

[608] M. SPURI, « Analysis of deadline scheduled real-time systems », *rapport de recherche n°2772*, Inria, janvier 1996.

[609] M. SPURI, « Holistic analysis for deadline scheduled real-time distributed system », *rapport de recherche n°2873*, Inria, avril 1996.

Divers

[610] E. ANCEAUME, G. LE LANN, P. MINET, « Étude des propriétés de tolérance aux fautes pour la fonction trains-trappes d'un système atterrisseur », 1996, programme GENIE, 10 p.

[611] G. LE LANN, P. MINET, « Analyse du comportement du CONTRONIC-E », Rapport d'avancement des travaux, 30 juin 1996, lot 1 du marché IPSN 4040 6B 024 870/SH, 36 p.

[612] G. LE LANN, P. MINET, « Analyse du comportement du CONTRONIC-E », Rapport final, 30 septembre 1996, lot 2 du marché IPSN 4040 6B 024 870/SH, 47 p.

8 Abstract

Proof-based System Engineering for Computing Systems and Real-Time Distributed Fault-Tolerant Computing are the areas considered.

The project investigates those algorithmic and methodological issues that arise with mission-critical, complex, computerized applications that may require certification.

Requirements of logical safety, liveness, timeliness and dependability, that are inevitably associated with such applications, can only be met with Real-time Distributed Fault-tolerant computing Technology, hence the "TRDF" acronym.

Research work is aimed at breaking new ground in the areas described below.

1) Refinement of a proof-based SE method

State-of-the-art in Computer Science cannot be transferred to users/technology providers unless embedded in a method that can be used by engineers. Furthermore, it is being recognized that the lack of a method for correctly and provably designing and dimensioning mission-critical, complex, computer-based systems is the main reason why a growing number of major failures are being experienced by the industry.

The project has developed a proof-based Systems Engineering method based on models and TRDF algorithms. The TRDF method involves correctness proof obligations. A design correctness proof obligation consists in verifying whether a given design (algorithms + models) solves a given problem. A dimensioning correctness proof obligation consists in verifying whether a valued correct design meets the physical requirements found in the application specification considered.

The specification of a system design/dimensioning that results from applying the TRDF method provably satisfies the specification of the application originally considered.

The TRDF method is being refined in view of potential transfer to external partners. In 1996, the TRDF method has been applied to three real problems, namely Modular Avionics, Safety Systems in Nuclear Power Plants, Safe Landing Gear. It has also been applied to the analysis of the Ariane 5 Flight 501 failure.

2) Composite TRDF algorithms

The goal pursued is to identify, prove and evaluate algorithms and protocols that are solutions to problems arising with real-time, fault-tolerant, distributed/concurrent computations and communications.

Issues of distribution are those arising in the presence of asynchronous parallel computations, with only partial knowledge of global system states. Real-time issues raise the obligation of proving that those timeliness constraints expressed in the specification of some application are always satisfied, for some feasibility conditions.

Fault-tolerance involves demonstrations that correct system behavior is maintained in the presence of given densities of partial failures, for given failure semantics.

For every algorithm/protocol studied, we establish such functions as upper bounds on response times and lower bounds on redundancy. Such functions are established using various techniques (e.g., graph theory, adversary arguments, calculus in (max, +) algebra) and considering deterministic adversaries. We also seek to express distance to optimality (the concept of optimal distributed on-line decision

making still is a fundamental research issue). In some instances, we establish that problems have no deterministic solutions.

Examples of results we have established are :

- extensive comparison in terms of complexity (of algorithms, of feasibility condiitons) and efficiency (of algorithms) between fixed - priority scheduling and deadline - driven scheduling,

- upper bounds on response times and feasibility conditions for real-time transactional applications over distributed client-server architectures (Stock Markets, Reservation, Air Traffic Control),

- feasibility conditions for a hybrid off-line/on-line scheduling algorithm aimed at modular avionics systems in the presence of failures.

We also have continued investigating the Asynchronous/Partially Synchronous Group Membership and Consensus problems.