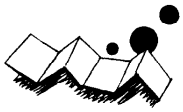
解　説

ヨーロッパにおける高信頼計算†



J. C. Laprie†† 井　原　廣　一††† 訳

まえがき：　ヨーロッパはパイオニアとしてやってきた．しかし・・・

最近の報告〔STR 81〕では高信頼 (Survival) システムの市場は来たる8年間に平均年 65% の率で成長し，1980年の7,500万ドルから1987年には26億ドルの世界的市場になるであろうと予測しており，耐故障 (fault tolerant) 計算，もっと一般的には高信頼 (dependable) 計算は，今や現実のものとなっている．

このようにフォールトトレラント計算が商業上の現実性をもってきたのは，直接あるいは間接的に，多大な科学分野での努力のためである．この論文での主題は“ヨーロッパはその中でいかなる地位にあるか”である．

A. Avizienis も述べたように，多くの科学，技術分野でヨーロッパはパイオニアとなってきた．

文献〔AVI 79〕によれば，

- フォールトトレラント計算の思想は，Dr. Dionysius Lardner が the Edinburgh Review に出版した論文 “Babbage's Calculating Engine” に書いた1834年にまで逆のぼる．

“計算の過程で発生するエラーについて，最も確かで有効なチェックは，同じ計算を分離して独立した計算機で行わせることであり，そしてそれらの計算機が異なる方法で計算をすれば，このチェックはいっそう決定的となる．

文献〔AVI 78〕によれば，

- 世界で（たぶん）最初のフォールトトレラント・ディジタル計算機は，プラハ（チェコスロバキア）で1950～1954年に設計された，SAPO であった．

しかし，フォールトトレランスのため慎重に設計されている，現在の商用汎用計算機システムに，Tandem Non-stop, August Can't Fail, Stratus（以上アメリカ）や，ACOS, HIDIC（日本）がある．

本論文はこのことにもかかわらず，ヨーロッパが学問としての高信頼計算の出現に重要な影響を及ぼしたことを示すのが目的である．この目的を達するため，ヨーロッパのすべての努力を列挙するのではなく（そのような仕事が可能としても一可能とは思われぬが一退屈であろうし，許されるページ数を超えると思うので），私が重要であると信ずることに焦点を合わせることにする．

1. 概　要

高信頼 (dependable) 計算は，システムが期待されている任務を果たす能力と定義できる．学問上，高信頼計算に見られる主要な2つ（実際はまざりあった）の柱は以下のとおりである．

（1） Procurement（構築）

- fault-avoidance（故障回避）：どうすれば最善の計算システムの仕様をつくり，設計し，実現利用できるか．
- fault-tolerance（耐故障）：どうすればシステムが（明らかな）エラーの存在するときにも任務を遂行し続けられるか．

（2） Validation（確実性）

- オフ・ライン：システムが任務を果たすことができるか．
- オン・ライン：どの程度まで，システムは任務を果たすことができるか．

構築に関しては，故障回避は一般にコンピュータサイエンスから分離するのが困難であるので，耐故障に局限して話をすすめる．しかし Dijkstra, Hoare, Parnas, Wirth〔DAH 72, DIJ 75, PAR 75, WIR 75〕のような著名な人々の影響が強くあることを述べておかねばならない．

確実性に関しては，どちらの観点も，明らかに高信頼計算の部分であり，そのつもりで述べる．

† Dependable Computing in Europe by Jean C. LAPRIE (Centre National De La Recherche Scientifique, Laboratoire d'Automatiqve et d'Analyse des Systems 1) and Hirokazu IHARA'S translation (Systems Development Laboratory, Hitachi Ltd.).

†† 国立科学研究院システム自動解析研究所
††† 日立製作所システム開発研究所

2. フォールトトレランスの構築

フォールトトレランス構築の研究は，扱われる故障の種類により，次のように述べることができる．

（1） 物理的（内的にしろ外的にしろ）要因で起るもの．

（2） 人為的（設計，相互作用，メンテナンス）要因で起るもの．

いくつかの抽象化レベルで計算システムを，暗黙的にか，明らさまにかを問わず構造化することと，人為的要因によるエラーの方が一般に物理的要因によるエラーよりも，より高度の抽象化レベルで観測されるということから，人的操作により生ずる故障に耐える対策は，設計不良による故障に耐える潜在能力をもっており，設計，物理的要因の故障にそれぞれ適用される，ということを，以後の議論で留意しておいて欲しい．

2.1 物理的故障に対するフォールトトレランス

フォールトトレランスにおける第一歩は自動誤り検知である．これを論理レベルで実現したものが，自己診断回路に対応する．〔MIN 67, CAR 68〕に報告されている先駆的な研究以来，ヨーロッパにおいて主として貢献してきたのはフランスであった．

- k-out-of-n コードを全般的に自己診断回路の構成に適用する研究．〔DIA 74〕
- 自己診断の概念を，a）8080 マイクロプロセッサ，RAM, ROM などの LSI 回路を使用したマイクロコンピュータ〔MOR 76〕と，b）直接マイクロプロセッサ〔CRO 80〕へ組み込み実装した例．

さて，誤りの検出，回復あるいは誤りのマスクによる効果的なフォールトトレランスに関して，ヨーロッパで最もよく知られた長期的研究としては次のものがあげられる．

- 航空，宇宙への応用としてフランスの初期の MECRA〔MAI 71〕やもっと最近の COPRA〔MER 76〕が実現している．しかし今のところ試作段階にとどまっている．
- 電子交換システムとして，イタリアの DST 1, DTN 1, AFDT 1〔BEL 76〕が実用化されている．

アーキテクチャ的な概念のほかに，これらの研究は次のような広範囲の興味ある結果をもたらした．

- CORPA における一時的（間けつ的 and/OR 過渡的）フォールトトレランスのための，再実行アルゴリズム．〔MEL 76〕
- UDET 7116（イタリア ESS の制御用計算機）におけるチェッカの自己試験周期の最適化．〔MOR 78〕

現在フランスで，保護システム（ここでは原子力プラントなどにおいて監視をしたり，事故の波及を防ぐための計算機システムを指す．）において特に重要な潜在故障を無くすための興味あるアーキテクチャの考え方が開発されている．この SPIN システムは，データ処理において，従来の 2-out-of-3 でなく 2-out-of-4 の多数決論理をとっており，操作器への指令信号についてフェイルセーフ・ロジックを採用している．〔REM 82〕．それは 1 つのユニットが定期点検中でも必要なサービスの連続性を保つためである．

以上に述べた研究は，すべて内部の物理的エラー（物理的内部要因によるものや，電磁的に誘因された物理的外部要因による故障）に関するものである．機械的な衝撃による物理的外部エラーに関しては，あまり文献がない．* この“フォールト・アンド・ダメージトレランス”の例は RHEA システムにみられる．〔POW 78〕

2.2 人為的要因のフォールトトレランス

物理的エラーに対するフォールトトレランスは確かに実用上重要であるが，それとは別に，多少とも複雑なシステムにおいては，設計エラーがそのままシステムの寿命を終えるまで残っているという確固たる信念に基づいて，重要な進歩がなされてきている．（序論で述べた 19 世紀の文献があるにもかかわらず，数年前においてすらこの信念が万人によっては受け入れられてはいなかったというのは面白い．たとえば〔MIL 75〕参照）設計エラーに対するフォルトトレランスは**機能的分散**の考え方に基づいている．ヨーロッパにおいてはイギリスがこれに力を入れており，“リカバリ・ブロック”という用語でよく知られている．〔RAN 75〕すなわち，ハード機器のスペアを用意するのと同じ方式の（または，その代わりをする）ソフトウェア・フォールトトレランスである．リカバリ・ブロックの働きは次のような考え方に基づいている：すなわち

交代動作する時，ブロックの変数のアップデイトを制限するための“リカバリ・キャッシュ”として知られているメカニズム〔LEE 79〕，リカバリ・ブロックを並行処理に応用するさい，簡単な概念的枠組を与

---

* その理由はおそらく軍事的応用に関係しているからである．注目に値する例として，〔SMI 75〕で示されているように，フランス Le Bourget でのトルコ航空 DC-10 の事故がある．

える，“アトミック・アクション”の考え方の採用．〔RAN 78〕

リカバリ・ブロック法の一番の弱点は受け入れテストである．現在でも，形式的な構成がなされようとしてはいるが，興味深いものはあるものの，いまだ不完全である．〔KOP 75〕

設計エラーに対する，もっと限定された（耐えうる故障のレベルの意味で），しかし実用上非常に重要な方法は**例外処理**である．このテーマに関しては，概念的にも〔CRI 80〕，実用面でも〔ICH 79〕ヨーロッパにおける仕事が確かにすぐれている．ADA がフランスのチームによって設計されたことをここで思い出して欲しい．

最後に，実行時に設計エラーを検出するための機能的分散の興味ある応用について述べようと思う．それは，“監視者”の概念で〔AYA 79〕，それによると2つの異なった抽象化レベルのソフトウェアバーション（すなわち，プログラムとその仕様書）が並行して動き，比較される．

相互作用による故障については，保護領域として知られる面白い考え方がある．この考え方は，誤りの検出と誤りの隔離を可能にするもので，広く研究されてきた．ヨーロッパにおける最も有名な実用化例は，イギリス Plessey 250〔ENG 24〕と CAP〔NEE 74〕計算機である．もっと最近では，ISAURE システムの設計が似たような考え方に基づいて進められている．〔BOI 78〕

3. オフライン確認

2章での問題として少々言及したが，オフライン確認とは，システムが運転前に故障（設計的と物理的あるいはそのどちらか）がないことの実証に関するものである．

W. C. Carter によれば，3つの互いに補間的なアプローチが考えられる；テスト，証明，および故障挿入に関する不良挿入テスト（mutation testing）である．〔CAR 82〕

3.1 テ ス ト

テストは故障を見つけることを目的とする．したがって，適当なテスト作業には準備として，捜している故障の種類の定義が必要である．単純な手順書による場合を除いては，テストは完全に N-P（非決定論的多項式）な問題である．それゆえに，テスト効率の適用範囲の評価が，特に必要になる．このことは公式モデルに基づくこの分野の研究の必要性を明らかにしている．

テストに関して行われている研究は，取り扱う故障の種類よりも，2つの抽象概念レベルすなわちハードウェアとソフトウェアに従って述べられる．第2章と違う点は，回路レベルのテスト対事前に計算した結果の予測値が，物理上かつ設計上の故障両方を検出する能力を持っているという事実である．

計算機についての現存の2つの見方，すなわち構造的（計算機はサブシステムの集まりとして見られる）あるいは関数的（計算機は計算アルゴリズムの手段として見られる）な見方に対応したテストアプローチを基準にして述べる．

テストに関する非常に多くの出版物の中で，上に述べた基準を満足するソフトウェアおよびハードウェアについての出版物のみを引用する．

ハードウェアレベルでは，構造的テストは現代の技術によって現われた複雑さに対応できるとは思われない．したがって，ここ何年かの間は，関数的テストの方向に研究は進められよう：

- フランスでは，ランダムテストが数年にわたって成果を上げてきている．〔RAU 71-TEL 74-DAV 80〕
- 決定論的なテストに関しては，マイクロプロセッサの ALU〔ROB 80〕あるいは RAM メモリ〔MAR 80〕の様な特定のサブシステムについてなされた研究についてのみ掲げておく．

現代の LSI 回路の複雑さに対処するため，かなり以前に，簡単なテストのできる設計の考え方が紹介された．ドイツでは，可能な技術のうちで，シフトレジスタによる scan-in scan-out 法が特によく研究された〔GRA 80〕．

現在までのソフトウェアのテストは，ほとんどが構造的アプローチであった．ヨーロッパでのこの方向のすぐれた研究として〔GEI 79〕がある．革新的で興味あるアプローチは，文法より導かれる．コンパイラテスト用のテストパターンの生成である．これは，論理回路のランダムテストに近いものである〔HOU 76〕．

ソフトウェアのテストは，ソフトウェアの品質評価と切っても切り放せないものである．これについては2つの非常に面白いモデルがイギリスで出現している．

- ソフトウェアの信頼性については，〔LIT 80〕に報告されている研究とそれに関係する文献が，私の

意見では，本質的な問題について最もよく書かれていると思う．

● ソフトウェアの複雑さについては，〔WOO 79〕に紹介された評価尺度が，以前に定められた尺度の不備を克服する点でたいへんおもしろいと思われる．

3.2 証　明

故障がないということを証明するのが目的である．現在，代数的アプローチとグラフ的アプローチの2つの方向で研究されている．

代数的アプローチとは2つの異なる抽象化レベルでの記号実行に基づいており，記号(公式)実行の開始と終了点における中間レベルの公式的な関係は正しいとするというものである．ヨーロッパでは，多くのチームがこの分野に関する研究を行っているが，アメリカに比べると，得られた結果は応用にはほど遠いものである．しかしイギリスでなされた研究〔BUR 69 & 77〕の概念上の影響は，ポーランドでマイクロプログラム検証用に設計された MIDDLE〔DEM 78〕という言語とともに特筆に値する．

グラフ的アプローチは，主にペトリネットに基づいており，代数的アプローチとは逆に，ヨーロッパがリードしている．もっともペトリネットが本来もっている検証能力は元来，アメリカで提唱されていたことである．

フランスやドイツではマークなしのペトリネットでモデルを検証するという重要な理論的進歩が得られている．すなわち不変埋没〔LAUT 74〕，縮約〔BER 79〕，そして段階的精緻化〔VAL 79〕の研究がそれである．これらの概念を統合したのが実験的会話型ツール OGIVE〔PRA 79〕である．

ペトリネットは並列処理のモデリングには適しているが，代数的アプローチと比較した場合，2つの不利な点がある．すなわちモデル内にデータを統合できない点（もっともこの点は，たとえばプロセスコントロールでのローカルコントローラーのような，ある特定のアプリケーションには有利になることもある．）そして，異なる抽象化レベル間に公式的な関係がないという点である．最初の問題については，非常に複雑なシステムに対して満足のゆく解は得られないと受容されているが，2番目の問題に対しては有望な結果が得られている．〔AND 81〕

なお，最近の高信頼計算に対する故障回避的方向，また仕様からプログラムへの連続的変換に基づく検証容易な設計については本論文では割愛する．

3.3 故障挿入に関する異常試験 (Mutation Test)

試作機への故障挿入がシステムの試験での不足を補うことができるという事実にもまして，異常試験はフォールトトレラント能力の検証や，故障範囲の正確な把握のための，現在唯一の合理的方法であり，将来，非常に重要なものになるだろう．しかし，この問題に関する研究では，機能的な実験を扱ったものしか発表されていない．〔MAI 71, DEC 80〕

4. オンライン確認

オンライン確認の目的は，稼動期間中の計算機システムのふるまいを定量化し，エラーが出現したときの保守や故障原因追求の見通しを明らかにすることである．

この定量化によってシステムの信頼性の尺度（信頼性，可用性，性能，または経済的に見た信頼性としての厳密な信頼性）を評価することができる．データ収集は興味深く，有用ではあるが（特にシステムの資源に関する基本的なデータを補強するためには，システムの信頼度が，仕様で指定された目標と一致するかどうかという古典的なチェックにとどまり，それをこえたシステムレベルでの評価という現在の目的を満たすことはできない．すなわち設計の各段階において，必要なデータを集める方法を選ぶことが必要であり，そうすればデータは確証のための助けとなる．

以上のことから，事前評価（すなわち稼動以前の）が必要となる．それはシステムのモデルに基本的なデータ（エラーの出現，保守，そして原因追求に関する）を挿入し，信頼度を得るためにその反応を見ることもできる．

挿入する基本データの性質（解析的かインパルス的か）と，システム表現の性質（モデルか試作機か）によって，システムの事前評価には3つの方法がある．モデリング，シミュレーション，そして観測である．先に述べたように異常試験は非常に有望ではあるが，依然として観測のときにだけ実施されている．

シミュレーションは近年，強力なツールとして生まれた．例として，フランスの SEGMA〔TRO 75〕があるが，過大な CPU 時間を必要とする．そこで最近ではモデリングが最も実際的なアプローチであるとされている．

フォールトトレラント計算機システムの評価のために行われてきたモデリングの研究に限れば，フランスでは確実な努力がなされてきており，マルコフモデル

に基づいたプログラム SURF〔COS 81〕を生んでいる．SURF は，

- 信頼性の統一的な考慮〔LAP 79〕
- 評価の信頼度を向上するためモデルの検証の重視〔LAP 80〕

を特徴としたり，より一般的な評価方式のツールである．また，ペトリネットとマルコフモデルとの間の補完性についての興味ある有望な研究も特筆に値する．〔BEY 81〕

なお種々の故障の，計算機の信頼性に及ぼす影響に関していえば，フランスやドイツでは数年前から稼動中のソフトウェアシステムをモデル化することに努力が注がれてきている．〔COS 78-LIT 76〕

5. 科学共同体としてのヨーロッパ

これまで述べてきたような研究に携わっている人々も他の分野の人々と同様に，国際的な技術交流を広く行っている．このことは，Fault-Tolerant Computing Symposia に応募した論文の，1971年における第1回目からこれまでの内訳を示した **表-1** によく示されている．

ヨーロッパ自体においては，EEC が EWICS (European Workshop on Industrial Computer System) を後援している．この EWICS には，いくつかの技術委員会がありそのうちの1つ，Safety and Security と称されている TC-7 が，高信頼計算についての研究を行っている．各国レベルにおいても，多くの国々が高信頼計算について研究しアイデアを交換するための体制をつくり出している．

- フランスでは，1977年に計算システムに重点をおいた高信頼システムに関する試験的なプロジェクト〔SUR 81〕が始められ，年に1度の研究集会で研究のサポート，調整を行っている．このプロジェクトは，高信頼計算について実際に共同研究が行われるもとと

表-1 高信頼システム国際会議 (FTCS) 論文一覧

	FTCS- 1 PASADENA (USA)	FTCS- 2 CAMBRIDGE (USA)	FTCS- 3 (PALO ALTO, USA)	FTCS- 4 (URBANA CHAMPAIN, USA)	FTCS- 5 (PARIS, FRANCE)	FTCS- 6 (PITTSBURGH, USA)	FTCS- 7 (LOS ANGELES, USA)	FTCS- 8 (TOULOUSE, FRANCE)	FTCS- 9 (MADISON, USA)	FTCS-10 (KYOTO, JAPAN)	FTCS-11 (PORTLAND, USA)	TOTAL
U. S. A.	32	32	47	21	29	27	35	29	19	32	31	334
FRANCE	3	2	4	3	17	9	6	7	6	13	12	82
JAPAN	1	2	2		1	2	2	3	2	16	6	37
ITALY					2	2	1	2	2	2	4	15
GREAT BRITAIN	1	1		1	1	1	2	2	2	1	2	14
GERMANY (WESTERN)					3	1	1	4		1	3	13
CANADA			1		1		1		2	2	2	9
ISRAEL			2		1	1	1					5
USSR					2	1		1		1		5
AUSTRALIA						1		1		1	1	4
POLAND					2	1				1		4
IRAN					1		1	1				3
NORWAY								1	1	1		3
CHINA										1	1	2
CZECHOSLOVAKIA								1			1	2
SWEDEN		1							1			2
BELGIUM					1							1
HUNGARY											1	1
MALAYSIA										1		1
TURKEY							1					1
YUGOSLAVIA					1							1
TOTAL	37	38	56	25	62	46	51	52	35	73	64	539

なっている.

● チェコスロバキアでは，約100人に及ぶ科学者たちが年に数回集まり，また国際会議(Fault-Tolerant Systems and Diagnostics)が，チェコスロバキア，ポーランドで1度ずつ開かれている.

● 西ドイツは，1982年3月に，第1回の fault-tolerant computer system に関する会議を開催した.

● イタリアは，1983年に開かれる第13回 Fault-Tolerant Computing Symposium の主催国となっている.

結　論

高信頼計算は疑いなく，先進国において情報処理科学の発展のために払われている研究の成功の鍵となるものである.

科学的観点から見ると，主となる問題は，現在及び未来のコンピュータシステムの複雑さを完全に会得する，ということにあるといえる．この複雑さによって生ずる多くの問題のうちでアーキテクチャの面から見て最も挑戦的なものは，コンピュータシステム（少なくとも機能的な意味で分散化されるであろう）の設計では，故障していることがはっきりしているサブシステムであっても，その定常運転状態においては構成要素である，という事実を考えに入れねばならないという事である．しかしシステムの複雑さのために，故障に対する許容性を前もって定めておくという方策では不十分であり設計者は，自己適応と呼べるような人工知能的な機構を取り入れねばならないであろう．この方向は〔GOL 75〕に述べられているが，現在までほとんど注目を受けなかった〔HEL 78〕．1つの示唆として，これらの研究のための興味深い体制は，現在フランスとアメリカで研究されている．“スーパコンピュータ”，あるいは，最近日本で開かれた“第5世代コンピュータ”会議との連携ではあるまいか.

しかし，最も問題になるものは，**ユーザ**が計算機システムの稼動について持たねばならない信頼感でありしたがって，**確認** (Varidation) である．私は次のことを強調したい．この必須の信頼感は，それが伝統的に求められるようなシステム（故障が破滅的結果につながりえるような）に限らず，人間とコンピュータとの（情緒的な？）関係が深まっているために（日常の情報生活において相互結合されているシステムのおかげで，どのような端末のユーザでも全体システムを信頼しなければならない），情報科学の全ての適用分野において，長い目で見ればより重要でさえある．ヨーロッパで行われている研究を知ることにより（ここであげたのは，その一部であるが），読者がこのヨーロッパの寄与について理解を深めていただけることを希望する.

謝辞 私は，この稿の執筆の機会を与えられた情報処理学会関係者，特に野口教授および当麻教授，翻訳を引き受けてくれた井原氏に深甚の感謝の意を表し，また A. Costes 氏の有益なコメントとご示唆に，J. P. Blanquart, J. C. Rault および R. Valette 諸氏のご協力に感謝する.

参 考 文 献

〔AND 81〕Andre, C.: Use of Behaviour Equivalence in Place-Transition Nets Analysis, 2nd European Workshop on Application and Theory of Petri Nets, Bad Honnef, West Germany (Sept. 28-30, 1981).

〔AVI 78〕Avizienis, A.: Fault-Tolerance: The Survival Attribute of Digital Systems, Proceedings of the IEEE, Special Issue on Fault-Tolerant Digital Systems, Vol. 66, N° 10, pp. 1109-1125 (Oct. 1978).

〔AVI 79〕Avizienis, A.: Toward a Discipline of Reliable Computing, IFIP Working Conference on Reliable Computing and Fault-Tolerance in the 1980's, London (Sept. 26-29, 1979).

〔AYA 79〕Ayache, J. M., Azema, P. and Diaz, M.: Observer: A concept for On-Line Detection of Control Errors in Concurrent Systems Proc. of the 9th Int. Symp. on Fault-Tolerant Computing (FTCS-9), Madison, Wisconsin, pp. 79-86 (June 20-22, 1979).

〔BEL 76〕Bellman, A.: Redundancy Design Approach Applied to Electronic Telephone Exchange Realizations, Proc. of the 6th Int. Symp. on Fault-Tolerant Computing (FTCS-6), Pittsburgh, pp. 9-16 (June 21-23, 1976).

〔BER 79〕Berthelot, G., Roucairol, G., Valk, R.: Reductions of nets and parallel programs, in Net Theory and Applications, Lecture Notes in Computer Science 84, Springer Verlag, Berlin, pp. 277-290 (1979).

〔BEY 81〕Beyaert, B., Florin, G., Lonc, P., Natkin, S.: Evaluation of computer systems dependability using stochastic Petri nets, Proc. of the 11th Int. Symp. on Fault-Tolerant Computing (FTCS-11), Portland, Maine, pp. 79-81 (June 24-26, 1981).

〔BOI 78〕Boi, L. and Michel, P.: An approach to a fault-tolerant system architecture, Proc. of

the 5th Int. Symp. on Computer, Palo Alto, California, pp. 123-130 (April 1978).

[BUR 69] Burstall, R. W.: Proving properties of programs by structural induction, The Computer Journal, Vol. 12, N° 1, pp. 41-48 (1969).

[BUR 77] Burstall, R. W. and Goguen, J.: Putting theories together to make specifications, Proc. of the 5th Int. Joint Conf. on Artificial Intelligence, Boston, Massachusetts, pp. 1045-1058 (August 1977).

[CAR 68] Carter, W. C., Schneider, P. R.: Design of dynamically checked computers, Proc. of IFIP '68 Con., Amsterdam, pp. 878-883 (1968).

[CAR 82] Carter, W. C.: The necessity for validation Collection of the transparencies presented at the Seminar on Computer System Function Validation, Toulouse, France, available from LAAS (Feb. 2-3, 1982).

[COS 78] Costes, A., Landrault, C. and Laprie, J. C.: Reliability and availability models for maintained systems featuring hardware failures and design faults, IEEE Trans. on Computers, Vol. C-27, N° 6, pp. 548-560 (June 1978).

[COS 81] Costes, A., Doucet, J. E., Landrault, C. and Laprie, J. C.: SURF, a program for dependability evaluation of complex fault-tolerant computing systems, Proc. of the 11th Int. Symp. on Fault-Tolerant Computing (FTCS-11), Portland, Maine, pp. 72-78 (June 24-26, 1981).

[CRI 80] Cristian, F.: Exception handling and software fault-tolerance, Proc. of the 10th Int. Symp. on Fault-Tolerant Computing, Kyoto, pp. 97-103 (Oct. 1-3, 1980).

[CRO 80] Crouzet, Y., Landrault, C.: Design of SC LSI circuits; application to a 4-bit microprocessor, IEEE Trans. on Computers, Vol. C-29, N° 6, pp. 532-537 (June 1980).

[DAH 72] Dahl, O. J., Dijkstra, E. W. and Hoare, C. A. R.: Structured programming, Academic Press, London and New-York (1972).

[DAV 80] David, R. and Thevenod-Fosse, P.: Minimal detecting transition sequences: application to random testing, IEEE Trans. on Computers, Vol. C-29, N° 6, pp. 514-518 (June 1980).

[DEC 80] Decouty, B., Michel, G. and Wagner, C.: An evaluation tool of fault-detection mechanisms efficiency, Proc. of the 10th Int. Symp. on Fault-Tolerant Computing, Kyoto, Japan, pp. 225-227 (Oct. 1-3, 1980).

[DEM 78] Dembinski, P. and Budkowski, S.: A verification design and description oriented microprogramming language, Proc. of the 1978 EUROMICRO Conf.

[DIA 74] Diaz, M.: Design of totally self-checking and fail-safe sequential machines, Proc. of the 4th Int. Symp. on Fault-Tolerant Computing, Urbana Champain, Illinois, pp. 19-24 (June 1974).

[DIJ 75] Dijkstra, E. W.: Guarded commands, non determinacy and formal derivation of programs, Communications of the ACM, Vol. 18, N° 8, pp. 453-457 (Aug. 1975).

[ENG 74] England, D. E.: Capability concept mechanisms and structure in system 250, Proc. of the Int. Workshop on Protection in Operating Systems, Rooquencourt, France, pp. 63-82 (Aug. 13-14, 1974).

[GEI 79] Geiger, W., Gmeiner, L., Trauboth, H. and Voges, U.: Program testing techniques for nuclear reactor protection systems, Computer, pp. 10-18 (Aug. 1979).

[GOL 75] Goldberg, J.: New problems in fault-tolerant computing, Proc. of the 5th Int. Symp. on Fault-Tolerant Computing (FTCS-5), Paris, France, pp. 29-34 (June 18-20, 1975).

[GRA 80] Grassl, G.:Design for testability, NATO Advanced Study Institute on Design Methodologies for VLSI, Louvain, Belgium (July 8-18, 1980).

[HEL 78] HELVIK, B. E.: An approach to optimal reconfiguration in dynamic fault-tolerant systems, Proc. of the 8th Int. Symp. on Fault-Tolerant Computing (FTCS-8), Toulouse, France p. 199 (June 21-23, 1978).

[HOU 76] HOUSSAIS, B.: Production systématique de tests commandés par une grammaire-application au test de compilateurs, Thesis, University of Rennes, France, in French (1976).

[ICH 79] ICHBIAH, J. D.: ADA reference manual, SIGPLAN Notices, Vol. 14, N° 6 (June 1979).

[KOP 75] Kopetz, H.: On the connections between range of variable and control structure testing, Proc. of the Int. Conf. on Reliable Software, Los Angeles, pp. 511-517 (April 21-23, 1975).

[LAP 79] Laprie, J. C.: Dependability modeling of computing systems, Invited State of-the-Art Report, IFIP Working Conference on Reliable Computing and Fault-Tolerance in the 1980's, London (Sept. 26-29, 1979).

[LAP 80] Laprie, J. C. and Medhaffer-Kanoun, K.: Dependability modeling of safety systems, Proc. of the 11th Int. Symp. on Fault-Tolerant Computing (FTCS-11), Kyoto, pp. 245-250 (Oct. 1-3, 1980).

[LAU 84] Lautenbach, K. and Schmid, H. A.: Use of Petri nets for proving correctness of concurrent process systems, Proc. of IFIP Congress

74, Stockholm, North Holland Publishing Co., pp. 187-191.

[LEE 80] Lee, P. A., Ghani, N. and Heron, K.: A recovery cache for the PDP-11, Proc. of the 9th Int. Symp. on Fault-Tolerant Computing, Madison, Wisconsin, pp. 3-8 (June 20-22, 1979).

[LIT 76] Littlewood, B.: A semi-Markov model for software reliability with failure costs, Proc. of 2nd Int. Symp. on Software Engineering, New-York, pp. 281-300 (April 20-22, 1976).

[LIT 80] Littlewood, B.: Theories of software reliability: how good are they and how can they be improved, IEEE Trans. on Software Engineering, Vol. SE-6, N° 5, pp. 489-500 (Sept. 1980).

[MAI 71] Maison, F. P.: The Mecra: a self-reconfigurable computer for highly reliable process, IEEE Trans. on Computers, Vol. C-20, N° 11, pp. 1382-1388 (Nov. 1971).

[MER 76] Meraud, C. and Browaeys, F.: Automatic rollback techniques of the Copra computer, Proc. of the 6th Int. Symp. on Fault-Tolerant Computing (FTCS-6), Pittsburgh, pp. 23-29 (June 21-23, 1976).

[MIL 75] Mills, H. D.: How to write correct programs and know it, Proc. of the Int. Conf. on Reliable Software, Los Angeles, pp. 363-370 (April 21-23, 1975).

[MOR 76] Moreira de Souza, J., Peixoto Paz, E. and Landrault, C.: A research-oriented microcomputer with built-in auto-diagnostics, Proc. of the 6th Int. Symp. on Fault-Tolerant Computing (FTCS-6), Pittsburgh, pp. 3-8 (June 21-23, 1976).

[MIN 67] Mine, H. and Koga, Y.: Basic properties and a construction method for fail-safe logical systems, IEEE Trans. on Electronic Computers, Vol. EC-16, N° 3, pp. 282-289 (June 1967).

[MOR 78] Morganti, M., Coppadoro, G. and Ceru, S.: UDET 7116, common control for PCM telephone exchange: diagnostic software design and availability evaluation, Porc. of the 8th Int. Symp. on Fault-Tolerant Computing (FTCS-8), Toulouse, France, pp. 16-23 (June 21-23, 1978).

[NEE 74] Nedham, R. M. and Walker, R. D. H.: Protection and process management in the CAP computer, Proc. of the Int. Workshop on Protection in Operating Systems, Rocquencourt, France, pp. 155-175 (Aug. 13-24, 1974).

[PAR 75] Parnas, D. L.: The influence of software structure on reliability, Proc. of the Int. Conf. on Reliable Software, Los Angeles, pp. 358-362 (April 21-23, 1975).

[ROW 78] Powell, D. R., Laprie, J. C., Romand, P. and Aleonard, C.: RHEA: a system for reliable and survivable interconnection of real-time processing elements, Proc. of the 8th Int. Symp. on Fault-Tolerant Computing (FTCS-8), Toulouse, France, pp. 117-122 (June 21-23, 1978).

[MAR 80] Marinescu, M.: Test fonctionnel de mémoire Vive á grande Couverture de pannes, Proc. of the 2nd Int. Conf. on Reliability and Maintainability, Perros-Guirec, France, pp. 31-37, in French (Sept. 1980).

[PRA 79] Pradin, B.: Un outil graphique interactif pour la vérification des systèmes à évolutions parallèles décrits par réseaux de Petri, Doctor-Engineer thesis, Paul Sabatier University, Toulouse, France, in French (Dec. 1979).

[RAN 75] Randell, B.: System structure for soft-tolerance, IEEE Trans. on Software Engineering, Vol. SE-1, N° 2, pp. 220-232 (June 1975).

[RAN 78] Randell, B., Lee, P. A. and Treleaven, P. C.: Reliability issues in computing system design, Computing Surveys, Vol. 10, N° 2. pp. 123-165 (June 1978).

[RAU 71] Rault, J. C.: A graph theoretical and probabilistic approach to the fault-detection of digital circuits, Proc. of the lst Int. Symp. on Fault-Tolerant Computing (FTCS-1), Pasadena, California, pp. 26-29 (March 1-3, 1971).

[REM 82] Remus, L.: Methodology for software development of a digital integrated protection system, EWICS TC-7 meeting, Brussels (Jan. 20-22, 1982).

[ROB 80] Robach, C. and Saucier, G.: Microprocessor functional testing, Proc. of the 1980 Cherry Hill Test Conference, pp. 433-443.

[SMI 75] Smith III, T. B.: A damage-and-fault-tolerant input-output network, IEEE Trans. on Computers, Vol. C-24, N° 5, pp. 505-512 (May 1975)

[STR 81] Strategic Inc.: Survivable computer systems to grow at 65% AAGR, 125 p.

[SUR 81] The Pilot-project SURF on systems dependability, Working report, Agency for Informatics, available from LAAS (Dec. 1981).

[TEL 74] Tellez-Giron, R. and David, R.: Random fault-detection in logical networks, Proc. of the Int. IFAC Symp. on Discrete Systems, Riga, USSR, pp. 232-241 (Sept. 30-Oct. 4, 1974).

[TRO 75] Troy, R., Paul, J. L. and Beaufils, R.: Evaluation globale de la sureté de fonctionnement des systèmes informatiques: SEGMA, Proc. of the 5th Int. Symp. on Fault-Tolerant Computing, Paris, France, pp. 98-103, in French (June 1975).

〔VAL 79〕 Valette, R.: Analysis of Petri nets by stepwise refinements, Journal of Computer and System Sciences, Vol. 18, N° 1, pp. 35-46 (Feb. 1979).

〔WIR 75〕 Wirth, N.: An assesment of the programming language PASCAL, Proc. of the Int. Conf. on Reliable Software, Los Angeles, pp. 23-30 (April 21-23, 1975).

〔WOO 79〕 Woodward, M. R., Hennell, M. A. and Hedly, D.: A measure of control flow complexity in program text, IEEE Trans. on Software Engineering, Vol. SE-5, N° 1, pp. 45-50 (Jan. 1979).

(昭和57年3月12日受付)