

JAERI-M
92-089

フォールトツリー解析コード
CUT-TDの使用手引

1992年6月

渡辺 憲夫・清田三紀雄*

JAERI-Mレポートは、日本原子力研究所が不定期に公刊している研究報告書です。

入手の間合わせは、日本原子力研究所技術情報部情報資料課（〒319-11 茨城県那珂郡東海村）あて、お申し込みください。なお、このほかに財団法人原子力弘済会資料センター（〒319-11 茨城県那珂郡東海村日本原子力研究所内）で複写による実費領布をおこなっております。

JAERI-M reports are issued irregularly.

Inquiries about availability of the reports should be addressed to Information Division Department of Technical Information, Japan Atomic Energy Research Institute, Tokaimura, Naka-gun, Ibaraki-ken 319-11, Japan.

© Japan Atomic Energy Research Institute, 1992

編集兼発行 日本原子力研究所
印刷 ニッセイエプロ株式会社

フォールトツリー解析コード
CUT-TDの使用手引

日本原子力研究所東海研究所原子炉安全工学部

渡辺 憲夫・清田三紀雄*

(1992年5月29日受理)

フォールトツリーからミニマルカットセットを算出し、頂上事象の発生確率を計算するための計算コードCUT-TDを開発した。本コードでは、従来のトップダウン法を改良し計算の効率化を図っている。CUT-TDコードにおける処理方法の特徴は、以下の通りである。

- ①論理和 "OR" ゲートあるいは論理積 "AND" ゲートが連続したレベルで存在する場合には、その繋がり全体を1つのゲートにまとめ、当該ゲートに対するカットセットを個々のゲートについての展開を行わずに求める。
- ②独立サブツリーを自動的に認識し、各独立サブツリーごとにミニマルカットセットを求めることによって計算の効率化を図っている。
- ③フォールトツリーの中間事象に対するミニマルカットセットを結合して頂上事象のミニマルカットセットを求めることが可能である。
- ④ミニマルカットセットに対する次数打ち切り及び確率打ち切りを指定することによってミニマルカットセットの数を減らし、計算効率の向上を図っている。
- ⑤ミニマルカットセットを求める段階で、展開を必要としない中間事象を指定することによって、ミニマルカットセットの数が少なくなり、効率的に計算を行うと共に解析者が分析しやすい形で結果を出力することができる。
- ⑥計算時間をプログラム内部で管理しており、CPUタイムオーバーによって計算が異常終了するのを防ぎ、その時点までの途中結果をファイル出力する。また、リスタート機能を有しており、この途中結果を入力として計算を実行できる。

本報告書は、CUT-TDコードの使用手引としてまとめたものである。

Users' Manual for Fault Tree Analysis Code: CUT-TD

Norio WATANABE and Mikio KIYOTA^{*}

Department of Reactor Safety Research
Tokai Research Establishment
Japan Atomic Energy Research Institute
Tokai-mura, Naka-gun, Ibaraki-ken

(Received May 29, 1992)

The CUT-TD code has been developed to find minimal cut sets for a given fault tree and to calculate the occurrence probability of its top event. This code uses an improved top-down algorithm which can enhance the efficiency in deriving minimal cut sets. The features in processing techniques incorporated into CUT-TD are as follows:

- (1) Consecutive OR gates or consecutive AND gates can be coalesced into a single gate. As a result, this processing directly produces cut sets for the redefined single gate with each gate not being developed.
- (2) The independent subtrees are automatically identified and their respective cut sets are separately found to enhance the efficiency in processing.
- (3) The minimal cut sets can be obtained for the top event of a fault tree by combining their respective minimal cut sets for several gates of the fault tree.
- (4) The user can reduce the computing time for finding minimal cut sets and control the size and significance of cut sets by inputting a minimum probability cut off and/or a maximum order cut off.

* Kozo Keikaku Engineering Inc.

- (5) The user can select events that need not to be further developed in the process of obtaining minimal cut sets. This option can reduce the number of minimal cut sets, save the computing time and assist the user in reviewing the result.
- (6) Computing time is monitored by the CUT-TD code so that it can prevent the running job from abnormally ending due to excessive CPU time and produce an intermediate result. The CUT-TD code has the ability to restart the calculation with use of the intermediate result.

This report provides a users' manual for the CUT-TD code.

Keywords: CUT-TD, Computer Code, Fault Tree Analysis, Minimal Cut Sets,
Improved Top-down Algorithm

目 次

1. はじめに	1
2. ミニマルカットセットの導出方法	5
3. CUT-TDコードの概要	8
3.1 CUT-TDコードの機能	8
3.2 処理方法における特徴	8
3.3 処理手順	11
3.4 使用上の制限事項	11
4. CUT-TDコードの入出力	15
4.1 入出力ファイル構成	15
4.2 入力データ	15
4.3 出力情報	18
4.4 エラーメッセージ	19
5. サンプル計算	34
5.1 FTA-Jコードとの比較	34
5.2 CUT-TDコードの性能確認	34
6. おわりに	42
参考文献	
付録 CUT-TDコードの構成	

Contents

1. Introduction	1
2. Methods for Obtaining Minimal Cut Sets	5
3. Brief Description of the CUT-TD Code	8
3.1 Function Overview	8
3.2 Features in Processing Techniques	8
3.3 Processing Flow	11
3.4 Restrictions for Using the CUT-TD Code	11
4. Input and Output of the CUT-TD Code	15
4.1 Input/Output Files	15
4.2 Input Data	15
4.3 Output	18
4.4 Error Message	19
5. Sample Calculations	34
5.1 Comparison with the FTA-J Code	34
5.2 Verification of the CUT-TD Code	34
6. Summary	42
References	43
Appendix Program Organization	44

1. はじめに

原子力発電プラントの確率論的安全評価 (Probabilistic Safety Assessment; PSA) においては、炉心が損傷し大量の放射性物質が格納容器から環境に放出されるような苛酷な事故を対象に、その発生頻度とそれがもたらす影響の大きさ (公衆の被曝量や財産損害の程度等) を推定し、プラントが公衆に及ぼす「リスク」を評価する。炉心損傷に至る事故シーケンス (即ち、炉心損傷事故シーケンス) の発生頻度の評価は、イベントツリー及びフォールトツリーを組合せることによって行われる。イベントツリーでは、事故の発端となる事象 (起因事象) に対して、事象の拡大防止に必要な機能・系統 (主として安全系) の作動/不作動を考慮して炉心損傷事故シーケンスを定義する。フォールトツリーでは、イベントツリーで採り上げた各種の機能・系統の機能喪失確率 (アンアベイラビリティ)、あるいは、イベントツリーで定義した炉心損傷事故シーケンスの発生頻度を求める。イベントツリー、フォールトツリー共、解析対象とするプラントのシステム構成に依存するため、作成にあたっては解析者による試行錯誤的な作業が必要となり多大な労力を要する。

原研では、この種の労力を軽減し効率よくイベントツリー解析及びフォールトツリー解析を行うための計算機ソフトウェアの開発を進めている。イベントツリー解析については、パーソナルコンピュータ上で稼動し対話形式でイベントツリーの作成及び定量化を行うための計算プログラム ETAP (Event Tree Analysis Supporting Program) を作成した⁽¹⁾。また、フォールトツリー解析については、1983年に、アンアベイラビリティの点推定計算及び不確実さ解析 (即ち、機器故障率や人的過誤率に存在する不確実さが頂上事象の発生確率にどのように伝播波及するかを調べるための解析) と、ミニマルカットセット (即ち、フォールトツリーの頂上事象が発生するに必要なかつ十分な要因あるいはその組合せ) の導出を一括して行うことを目的として、米国で開発された幾つかの計算プログラムを統合し、総合的な計算コード FTA-J を作成した⁽²⁾。しかしながら、FTA-Jコードについてプログラムの改良や機能拡張等を行う際には、プログラムサイズが大きく、また、設計思想の異なるプログラムを有しているため、改良作業が複雑なものになる。

この問題点を解消すると共に、より効率的にフォールトツリー解析を行えるよう、現在、新たな計算コードパッケージ REFT の開発・整備を行っている。Fig. 1.1 に、REFTコードパッケージの概要を示す。REFTの開発にあたっては、以下の項目に主眼をおいている。

- (1) 機器故障率データベースとの結合
機器故障率データベース RECORD⁽³⁾ と結合し、機器故障率データの検索及び統計処理の結果を容易に利用できる。
- (2) 対話形式によるフォールトツリーの作成
フォールトツリーの作成を、パーソナルコンピュータ上で対話形式によって行うことが可能であり、フォールトツリー作成時の試行錯誤的な作業を効率よく行える。
- (3) ミニマルカットセット導出ロジックの変更
ミニマルカットセットを導出するための計算方法にトップダウン方式を採用することによって、解析時間の短縮化を図る (FTA-Jコードで採用されている計算ロジックはボトムアップ方式である)。なお、トップダウン方式及びボトムアップ方式については2章に具体的に記述する。

(4) 時間依存アンベイラビリティの計算

残留熱除去系のように、事故の初期の段階では炉心への冷却材注入を、また、その後は炉心からの崩壊熱を除去することを要求される系統について、フェーズドミッション（系統の使命が時間的に変化すること）を考慮し、系統の機能喪失確率を時間依存で計算することが可能である。

(5) 頂上事象発生確率に対する不確実さ解析

フォールトツリーの頂上事象に対するミニマルカットセットを直接入力することによって、頂上事象の発生確率に対する不確実さ解析を行うことができる。なお、FTA-Jコードでは、ミニマルカットセットを基に確率計算式を生成する必要がある（例えば、ミニマルカットセットA、Bに対して、 $P(\text{top}) = P(A) + P(B) - P(A) * P(B)$ なる式を生成する）、さらに、この式をFTA-Jコードの1サブルーチンとして扱うため、計算に際しては同サブルーチンの編集結合が必要となる。

このうち、(1)については、大型計算機（FACOM M-780）上で作成したデータベースRECORDをパーソナルコンピュータに移植し、使用性の向上を図っている。(2)については、フォールトツリー作成支援プログラムPC-CREFTSの第1次版を作成し、現在使用手引の整備を進めている。(3)、(4)及び(5)については、それぞれ大型計算機上で稼動する計算コードCUT-TD、TD-BAM及びSPASM-Rの第1次版を作成した。

本報告書は、(3)の計算コードCUT-TDの使用手引としてまとめたものである。CUT-TDコードは、フォールトツリーからミニマルカットセットを求めるための計算プログラムであり、FORTRAN77（一部アセンブラ）で書かれ大型計算機FACOM M-780上で稼動する。同コードの機能は、以下の通りである。

- ①フォールトツリーの頂上事象に対して、ミニマルカットセットを求める。
- ②フォールトツリー上の機器故障や人的過誤等の素事象（以後、コンポーネントと呼ぶ）の生起確率（機器故障率や人的過誤率等）を用いて、各ミニマルカットセットの生起確率を計算すると共に、フォールトツリーの頂上事象の発生確率を計算する。
- ③不確実さ解析のための計算コードSPASM-Rの入力データファイルを作成する。

CUT-TDコードの開発に際しては、ミニマルカットセットを求めるための計算効率を向上させることに主眼を置き、処理方法に次のような工夫を施している。

- (1) 論理和"OR"ゲートあるいは論理積"AND"ゲートが連続したレベルで存在する場合には、その繋がり全体を1つのゲートにまとめ、当該ゲートに対するカットセットを直接求める。例えば、Fig. 1.2のフォールトツリーに示すように、下位レベルが全て"OR"ゲートあるいは"AND"ゲートであるゲートG2あるいはG3は、以下のように当該ゲートに対するカットセットを用いて再定義される。

$$G2 = E1 + E3 + E2 + E10 \quad (1.1)$$

$$G3 = E4 * E5 * E6 * E7 \quad (1.2)$$

そのため、ゲートG5、G6、G7、G9については展開処理が省略できる。但し、ゲートG4及びG8は、その下位レベルのゲートが同一タイプでないためこの処理は適用されない。

- (2) 独立サブツリー（他の部分と共通に使用するコンポーネントやゲートがなく独立している部分を指す）を自動的に認識し、各独立サブツリーごとにミニマルカットセットを求めることができる。これは、ブール演算則（吸収則や巾等則）の処理に要する計算時間を軽減するための工夫である。
- (3) 既にミニマルカットセットが求められている事象を中間事象（ゲート）として入力することができ、これによって、複数のミニマルカットセット情報を結合して頂上事象に対するミニマルカットセットを求めることが可能となる。
- (4) ミニマルカットセットを導出する段階で、次数による打ち切り、及び、生起確率による打ち切りが指定できる。これによって、頂上事象の発生確率に殆ど寄与しないようなミニマルカットセットを求めずに計算を実行することができる。

- (5) ミニマルカットセットを求める段階で、展開しないゲートを指定することができ、ゲートをコンポーネントと同等に扱う機能を有する。この機能は、解析者にとって不必要と思われるミニマルカットセットを求めることなく効率的に計算を行うことを目的とする。
- (6) 計算時間（CPUタイム）をプログラム内部で管理しており、計算時間の打切り（CPUタイムオーバー）によって計算が異常終了するのを防ぎ、その時点までの途中結果をファイル出力する。さらに、この途中結果を用いて再実行することができる（リスタート機能）。

なお、上記(2)及び(4)については、既に幾つかの計算コードで同様の方法が採用されている。

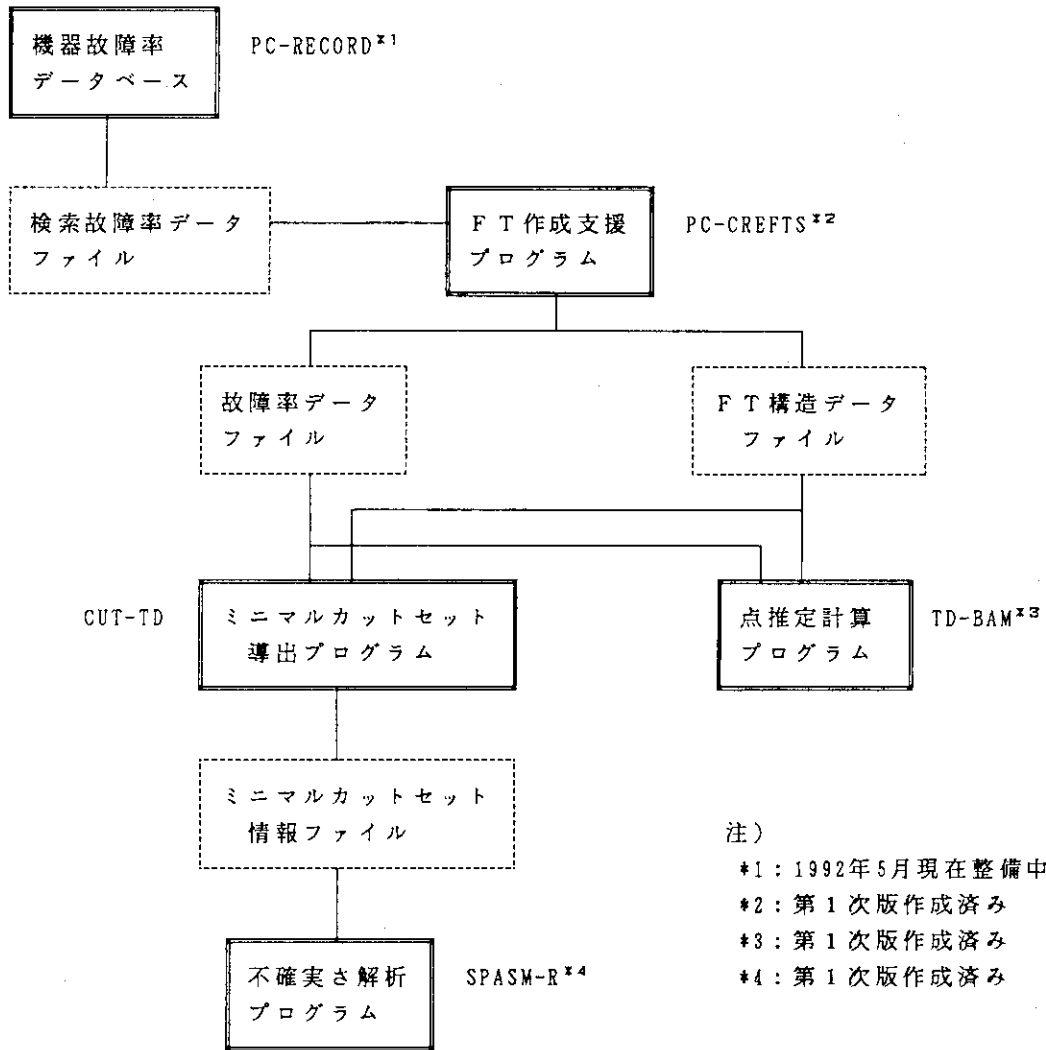
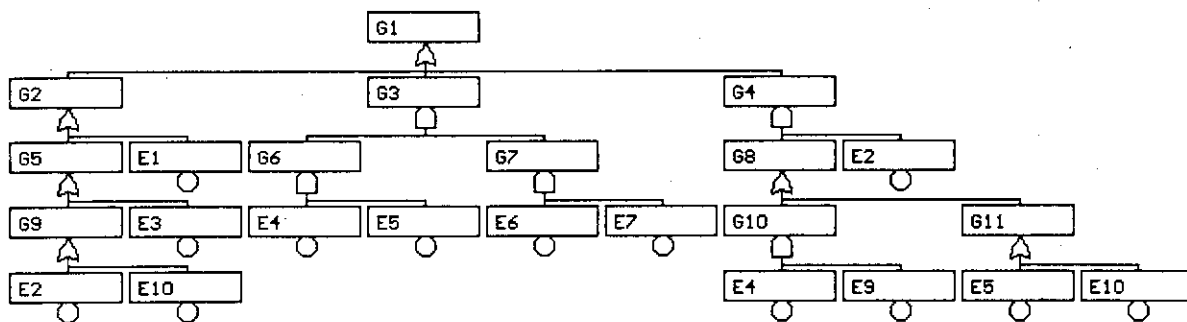


Fig. 1.1 フォールトツリー解析コードパッケージREFTの構成



- ・ゲートG2について、その下位レベルのゲートG5及びG9も"OR"ゲートであるため、これら3つのゲートを一括して扱う。
- ・ゲートG3について、その下位レベルのゲートG6及びG7も"AND"ゲートであるため、これら3つのゲートを一括して扱う。
- ・ゲートG4については、下位レベルに"OR"ゲートと"AND"ゲートが混在するため一括して扱うことはできない。

Fig. 1.2 同一タイプのゲートが連続したレベルで存在するフォールトツリーの例

2. ミニマルカットセット導出方法

ミニマルカットセット (Minimal Cut SET : MCS) を導出するための計算コードには、一般に、次の2つの方法が用いられている。

- (1) ボトムアップ方式
- (2) トップダウン方式

ボトムアップ方式は、P. Cahtterjeeによって提案された方法⁽⁴⁾で、WAMCUT⁽⁵⁾、MICSUP⁽⁶⁾等の計算コードに採用されている。先に原研で作成したFTA-Jコードには、WAMCUTコードが組込まれている。この方法では、フォールトツリーの末端にあるゲートをコンポーネントから成る論理式に展開する。次に、その上位の各ゲートについて論理展開を行い、その入力となるゲートの論理式を適用してコンポーネントから成る論理式に変換する。この段階で、ブール演算則（吸収則及び巾等則）を適用し論理式を簡略化する。ゲートの展開処理を順次上方のゲートに対して繰り返すことによってフォールトツリーの頂上事象に対するミニマルカットセットを導出する。

一方、トップダウン方式は、J. B. Fussel & W. E. Veselyによって提案された方法⁽⁷⁾で、SETS⁽⁸⁾、WAMCUT-II⁽⁹⁾等数多くの計算コードに適用されている。この方法では、フォールトツリーの頂上事象から下方に、ブール演算則に従って順次ゲートを展開していく。まず、頂上事象を、その入力となるゲート／コンポーネントから成る論理演算式に展開し、次に、その演算式を構成するゲートについて展開を行う。ゲートの展開に際しては、吸収則及び巾等則を適用し論理式を簡略化する。この処理を、最終的に頂上事象がコンポーネントだけから成る論理演算式になるまで繰り返すことによって頂上事象に対するミニマルカットセットを決定する。

以下に、Fig. 2.1に示すフォールトツリーを対象に、これら2つの方法による具体例を記述する。

ボトムアップ法では、まず、9つのゲート (G1~G9) のうちツリーの末端に位置するゲート (G6, G7, G8, G9) 各々について、論理式の展開が行われる。

$$G8 = X13 + X14 \quad (2.1)$$

$$G7 = X12 + X13 \quad (2.2)$$

$$G6 = \overline{X9} \cdot X10 \quad (2.3)$$

$$G9 = \overline{X9} \cdot X10 \quad (2.4)$$

次に、これらの展開式を用いて、その上位のゲート (G4, G5) が展開される。

$$G5 = G7 \cdot G8 \quad (2.5-1)$$

$$= (X12 + X13) \cdot (X13 + X14) \quad (2.5-2)$$

$$= X12 \cdot X13 + X12 \cdot X14 + X13 + X13 \cdot X14 \quad (2.5-3)$$

$$= \overline{X13} + X12 \cdot X14 \quad (2.5-4)$$

$$G4 = \overline{G6} \cdot X11 \quad (2.6-1)$$

$$= \overline{X9 \cdot X10} \cdot X11 \quad (2.6-2)$$

$$= (\overline{X9} + \overline{X10}) \cdot X11 \quad (2.6-3)$$

$$= \overline{X9} \cdot X11 + \overline{X10} \cdot X11 \quad (2.6-4)$$

さらに、その上位にあるゲート (G1, G2, G3) についても同様の展開を行う。

$$G3 = G4 + G5 \quad (2.7-1)$$

$$= (\overline{X9} \cdot X11 + \overline{X10} \cdot X11) + (X13 + X12 \cdot X14) \quad (2.7-2)$$

$$= \overline{X9} \cdot X11 + \overline{X10} \cdot X11 + X13 + X12 \cdot X14 \quad (2.7-3)$$

$$G2 = G3 \cdot G4 \quad (2.8-1)$$

$$= (\overline{X9 \cdot X11} + \overline{X10 \cdot X11} + X13 + X12 \cdot X14) \cdot (\overline{X9 \cdot X11} + \overline{X10 \cdot X11}) \quad (2.8-2)$$

$$= \overline{X9 \cdot X11} + \overline{X9 \cdot X10 \cdot X11} + \overline{X9 \cdot X11 \cdot X13} + \overline{X9 \cdot X11 \cdot X12 \cdot X14} \\ + \overline{X9 \cdot X10 \cdot X11} + \overline{X10 \cdot X11} + \overline{X10 \cdot X11 \cdot X13} + \overline{X10 \cdot X11 \cdot X12 \cdot X14} \quad (2.8-3)$$

$$= \overline{X9 \cdot X11} + \overline{X10 \cdot X11} \quad (2.8-4)$$

$$G1 = G2 + G3 + G6 \quad (2.9-1)$$

$$= (\overline{X9 \cdot X11} + \overline{X10 \cdot X11})$$

$$+ (\overline{X9 \cdot X11} + \overline{X10 \cdot X11} + X13 + X12 \cdot X14)$$

$$+ (\overline{X9 \cdot X10}) \quad (2.9-2)$$

$$= \overline{X9 \cdot X11} + \overline{X10 \cdot X11} + X13 + X12 \cdot X14 + \overline{X9 \cdot X10} \quad (9-3)$$

これに対し、トップダウン法では、下記のように、頂上事象から逐次展開する。

$$G1 = G2 + G3 + G6 \quad (2.10-1)$$

$$= (G3 \cdot G4) + G3 + G6 = G3 + G6 \quad (2.10-2)$$

$$= G4 + G5 + G6 \quad (2.10-3)$$

$$= G9 \cdot X11 + G7 \cdot G8 + G6 \quad (2.10-4)$$

$$= \overline{X9 \cdot X10 \cdot X11} + (X12 + X13) \cdot (X13 + X14) + \overline{X9 \cdot X10} \quad (2.10-5)$$

$$= (\overline{X9} + \overline{X10}) \cdot X11 + (X13 + X12 \cdot X14) + \overline{X9 \cdot X10} \quad (2.10-6)$$

$$= \overline{X9 \cdot X11} + \overline{X10 \cdot X11} + X13 + X12 \cdot X14 + \overline{X9 \cdot X10} \quad (2.10-7)$$

この例に示したように、ボトムアップ法では、ツリーの下から順に展開し、その結果を用いて上位のゲートに対して論理式を作成するため、全てのゲートについて展開処理を行い、さらに、コンポーネントだけから成る論理式に展開した後に吸収則や巾等則を適用する必要がある。これに対して、トップダウン法では、ゲートに展開した時点で吸収則や巾等則を適用することが可能であり、展開すべき項目を少なくすることができる。上の例では、式(2.10-2)の部分に示すように、ゲート同志で吸収則が適用され(G3・G4)の項がなくなるため、この項に関するゲートの展開処理が省略されることになる。このように、多くの場合には、トップダウン法の方が効率的にミニマルカットセットを導出することができる。但し、数多くのゲートに対してカットセットを求める場合、あるいは、フォールトツリー構造が比較的単純でコンポーネントが互いに独立であるような場合には、ボトムアップ法の方が効率的にミニマルカットセットを導出できる。

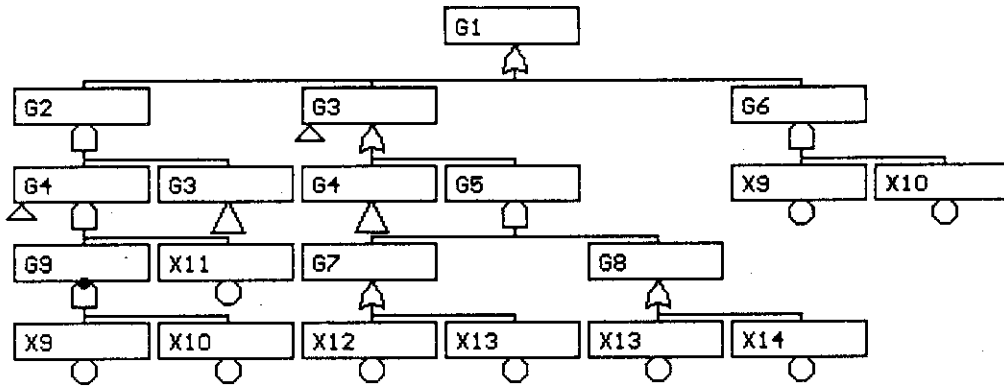


Fig. 2.1 サンプル・フォールトツリー

3. CUT-TDコードの概要

3.1 CUT-TDコードの機能

CUT-TDコードは、トップダウン法によりミニマルカットセットを求めるための計算コードである。本コードの機能は、以下の通りである。

(1) ミニマルカットセットの導出

解析対象とするフォールトツリーの頂上事象及びユーザが指定するゲートに対して、ミニマルカットセットを求める。

(2) ミニマルカットセットの生起確率計算

各ミニマルカットセットについてその生起確率 $P(MCS)_i$ を計算する。

$$P(MCS)_i = \prod P(\text{component})_j \quad (3.1)$$

但し、 $P(\text{component})_j$: ミニマルカットセット i を構成するコンポーネント j の生起確率

(3) 頂上事象発生確率の計算

フォールトツリーの頂上事象に対するミニマルカットセットの生起確率の総和をとり、頂上事象の発生確率 $P(\text{total})$ を計算する。

$$P(\text{total}) = \sum P(MCS)_i \quad (3.2)$$

(4) ミニマルカットセットの重要度計算

各ミニマルカットセットについて、その生起確率 $P(MCS)_i$ が頂上事象の発生確率 $P(\text{total})$ にどの程度寄与しているか（即ち、ミニマルカットセットの重要度）を計算する。

$$I(MCS)_i = P(MCS)_i / P(\text{total}) \quad (3.3)$$

但し、頂上事象の発生確率 $P(\text{total})$ は、上記(3.2)式で求めたものの他に、FTA-Jコード等で計算した値をユーザが入力データとして与えることも可能である。

(5) 不確実さ解析用の入力データの作成

ミニマルカットセット、及び、それを構成するコンポーネントの生起確率データを、不確実さ解析コード SPASM-R の入力データ形式に編集しファイルに書き出す。

3.2 処理方法における特徴

CUT-TDコードで用いたアルゴリズムは、計算効率の向上を図ることを目的に、従来のトップダウン法を改良したものである。主要な改良点は、以下の通りである。

(1) 同一タイプのゲートが連続したレベルで存在する場合の一括処理

従来のトップダウンアルゴリズムは、各レベルのゲート1つ1つに対して順次展開を行い、各ゲートごとにカットセットを求めている。これに対し、CUT-TDコードのアルゴリズムでは、“OR”ゲートあるいは“AND”ゲートが連続したレベルで存在する場合には、その繋がり全体を1つのゲートにまとめ、当該ゲートに対して、その下位レベルにある個々のゲートの展開を行わずに、カットセットを求める。例えば、Fig. 3.1に示すフォールトツリーを従来法とCUT-TDのアルゴリズムを用いて展開すると、次のような違いが生じる。

(従来法)	(CUT-TDアルゴリズム)
$G1 = G2 * G3$	$G1 = A * B * C * G3$
$G2 = G4 * C$	$G3 = D + E + G6$
$G3 = G5 + G6$	$G6 = F * G$
$G4 = A * B$	
$G5 = D + E$	
$G6 = F * G$	

この例から分るように、従来法では、フォールトツリー上に出現するゲート全て (G1~G6) に対して展開処理を行い、各ゲートごとにカットセットを求めるため、結局6回の展開処理を行うことになる。一方、CUT-TDコードのアルゴリズムでは、G1、G2、G4と3つのレベルで連続して"AND"ゲートが繋がるため、G2、G4については個別の展開を行わず、G1の展開の際に一括して展開処理を行うことになる。同様に、G5についても、その親ゲートG3と同一タイプのゲートであるため、G5の個別の展開処理が省略される。従って、CUT-TDではG1、G3、G6の3つのゲートについてのみ展開処理が実行されることになり、この処理に要する時間の短縮を図っている。

(2) 独立サブツリーの自動認識

トップダウンアルゴリズムを用いて直接フォールトツリーを展開すると、ミニマルカットセットを求める過程で、フォールトツリー上に出現する全てのゲート及びコンポーネントについて、ブール演算の吸収則及び巾等則を適用することが必要となり、多くの計算時間を要する。そこで、CUT-TDコードでは、計算時間の短縮を図るために、独立サブツリー (あるゲートの子孫となるゲート及びコンポーネントが、他の部分と独立しているもの) を識別し、吸収則及び巾等則を適用すべき対象を少なくしている。即ち、独立サブツリー全体を1つのコンポーネントと見做して (これを「疑似コンポーネント」と呼ぶ) フォールトツリーの展開処理を行うことが可能となり、この際、独立サブツリーの構成要素が他の部分で参照されないため、他の部分の構成要素との間で吸収則や巾等則を適用せずにミニマルカットセットを求めることができる。独立サブツリーの識別処理を行わない場合と、行う場合について、Fig. 3.2に示すフォールトツリーを例にとってミニマルカットセットを求める過程を以下に示す。

独立サブツリーの識別処理を行わない場合

このフォールトツリーに対して、独立サブツリーの処理を行わないでミニマルカットセットを求めようとすると、頂上事象G1に対する基本式は、10個の構成要素から成る式で表現され、この式を展開してミニマルカットセットを求める場合には、これら構成要素各々の間で吸収則や巾等則を適用する必要がある。

$$G1 = \{E1 + E3 * (E2 + E10)\} \\ + \{(E4 + E5) * (E6 + E7)\} \\ + \{(E8 + E9) * E2\}$$

その結果、以下に示す9個のミニマルカットセットが得られる。

$$E1, E2 \cdot E3, E3 \cdot E10, E4 \cdot E6, E4 \cdot E7, E5 \cdot E6, E5 \cdot E7, E2 \cdot E8, E2 \cdot E9$$

独立サブツリーの識別処理を行う場合

Fig. 3.2のフォールトツリーにおいて、ゲートG3以下の部分とゲートG8以下の部分は、他の部分で参照されていないため、独立サブツリーとして識別することができる。これらの部分を独立サブツリーとして識別し、疑似コンポーネントと見做してフォールトツリーの展開処理を行うと、頂上事象G1に対する基本式は、次のように、6つの構成要素からなる簡単な形となる。

$$G1 = \{E1 + E3 * (E2 + E10)\} + G3 + (E2 * G8)$$

この式を構成する6つの要素間で吸収則や巾等則を適用し基本式を展開することによって、以下に示すように、疑似コンポーネントを含む5個のミニマルカットセットが求まる。

$$E1, E2 \cdot E3, E3 \cdot E10, G3, E2 \cdot G8$$

さらに、G3とG8についての基本式を求めて、その結果を上記のミニマルカットセットに代入することによって頂上事象G1に対する最終的なミニマルカットセットを得ることができる。しかし、G3とG8の構成要素は、他の部分で参照されないため、G3とG8に対する基本式を代入する際には吸収則や巾等則を適用する必要はない。

このように、独立サブツリーの識別処理を行った場合には、吸収則や巾等則を適用すべき構成要素の数を減らすことができ、ミニマルカットセットを効率的に求めることが可能となる。

(3) ミニマルカットセットの結合処理

あるゲートに対して予めミニマルカットセットを求めておき、それを入力データとして与えることができる。これによって、ゲートに対する展開処理を省略することとなり、フォールトツリーの頂上事象に対するミニマルカットセットの導出を効率的に行える。特に、複数のフォールトツリーを結合したものについては、予め各々のフォールトツリーに対するミニマルカットセットを求めておくことにより、ツリーの展開処理を一切行わずに、複数のフォールトツリーを結合したものに対するミニマルカットセットを求めることができる。なお、CUT-TDコードでは、ミニマルカットセットを入力として与えるゲートを“MCS”ゲートと呼ぶこととする。

(4) ミニマルカットセット打切条件の採用

ミニマルカットセットを求める段階で、カットセットの次数による打切条件、及び、カットセットの生起確率による打切条件を指定することができる。これによって、ユーザ（解析者）の指定した条件に沿って、ツリーの展開処理を省略することが可能となり、ミニマルカットセットを効率良く求めることができる。

(5) 展開不要ゲートの採用

ツリーの展開が不要であると指定されたゲート（CUT-TDコードでは、「展開不要ゲート」と呼ぶ）をコンポーネントと見做してフォールトツリーの展開を行う。この結果得られるミニマルカットセットは展開不要ゲートを含む形で表現されるためミニマルカットセットの数が少なくなり、解析者が重要な機器故障の組合せを把握する上で分析しやすい形態で計算結果を出力することとなる。特に、大規模なフォールトツリーではミニマルカットセットの数が膨大となるため、その数を制限することは、結果のレビュー及び重要な機器故障の組合せの識別に要する労力の軽減に繋がるものと考えられる。なお、展開不要ゲートは、ユーザが任意に指定できる。この場合、展開不要ゲートの生起確率は自動的に1.0と設定されるが、これは、フォールトツリーの構造データを変更せずに、当該ゲートの下位レベルに存在するコンポーネントの生起確率がフォールトツリーの頂上事象の発生確率に影響を及ぼさないことを確認することを目的としている。言い換えれば、展開不要ゲートを指定してミニマルカットセットを求めることによって、当該ゲートを含むミニマルカットセットの頂上事象の発生確率への寄与がどの程度かを容易に把握できる。

(6) リスタート機能の採用

CUT-TDコードでは、計算時間をプログラム内部で管理しており、ユーザが指定し

た処理時間内にミニマルカットセットを全て導出することができない場合に、その途中結果をファイルに格納する。さらに、同ファイルを読み込み途中結果から計算を継続することが可能である。このリスタート機能を採用することによって、ユーザは上記(4)の打切条件を指定しなくても、途中結果を見て適当な段階で計算を終了させることができる。

3.3 処理手順

CUT-TDコードにおける処理は、Fig. 3.3に示すように、以下の6つのステップから構成される。

(1) 入力データの読み込み

フォールトツリーの構造データ、各コンポーネントの生起確率データ、打切り条件等の計算制御用データを読み込む。この際、MCSゲート及び展開不要ゲートを認識する。

(2) フォールトツリーの構造変換

コンビネーションゲートを"OR"ゲートあるいは"AND"ゲートによる表現に変換すると共に、"NAND"ゲートを"NOT"ゲートと"OR"ゲート(例えば、 $A \cap B \rightarrow A \cup B$)に、"NOR"ゲートを"NOT"ゲートと"AND"ゲート($A \cup B \rightarrow \overline{A \cap B}$)に変換する。また、同一のゲートタイプ("OR"ゲートあるいは"AND"ゲート)が連続したレベルで存在する場合には、その繋がり全体を1つのゲートにまとめ、当該ゲートのカットセットを求める。

(3) 独立サブツリーの認識

独立サブツリーとなるゲートを識別する。

(4) 独立サブツリーに対するカットセットの導出

独立サブツリーとして識別された部分の頂上事象となるゲートに対して、展開処理を行い巾等則及び吸収則を適用して、各独立サブツリーのカットセットを求める。

(5) 頂上事象に対するカットセットの導出

独立サブツリーとして識別された部分を一時的にMCSゲートと見做してフォールトツリーの頂上事象に対する展開処理を行い、巾等則及び吸収則を適用してカットセットを求める。

(6) 頂上事象に対するミニマルカットセットの導出(カットセットの結合)

MCSゲートに対して入力で与えられたカットセット、及び、上記の(4)で求めた独立サブツリーのカットセットを、上記(5)の処理結果(上記(5)の処理で求めたカットセットにはMCSゲートが含まれている)に代入することによって、フォールトツリーの頂上事象に対するミニマルカットセットを求める。なお、各独立サブツリーのカットセットを構成する要素は他のカットセットの構成要素と独立な関係にあるため、この処理において各独立サブツリーのカットセットを代入する際、吸収則及び巾等則の適用は省略される。

3.4 使用上の制限事項

CUT-TDコードの開発に際して、入力データの1つであるフォールトツリーの構造に関するデータに、FTA-Jコードと互換性を持たせることとしたため、以下のような使用上の制約を設定した。

- ・ゲート名及びコンポーネント名 : 8文字以内の英数字
- ・使用可能なゲートタイプ :
 - AND --- $a \cap b$
 - OR --- $a \cup b$
 - NOT --- $\neg a$
 - NOR --- $\neg (a \cup b)$
 - NAND -- $\neg (a \cap b)$
 - ANOT -- $a \cap (\neg b)$
 - ONOT -- $a \cup (\neg b)$
 - COM --- n out of m (2 out of 3, 2 out of 4等)
- ・各ゲートタイプに対する入力ゲート及び入力コンポーネントの総数 n :
 - AND --- $2 \leq n \leq 8$ 個
 - OR --- $2 \leq n \leq 8$ 個
 - NOT --- n=1個
 - NOR --- n=2個
 - NAND -- n=2個
 - ANOT -- n=2個
 - ONOT -- n=2個
 - COM --- $3 \leq n \leq 8$ 個

この他、CUT-TDコード特有の制約事項としては、以下のものがある。

- ・導出可能なミニマルカットセットの最大次数 : 10次
 - ・MCSゲートに入力として与えられるミニマルカットセットの最大次数 : 8次
 - ・解析対象フォールトツリーの規模 :
 - ゲート総数 --- 500個以内
 - コンポーネント総数 --- 500個以内
 - ・導出可能なミニマルカットセットの総数 : 20,000個
- 但し、ゲート総数、コンポーネント総数、及び、導出可能なミニマルカットセットの総数は、いずれもパラメータ文で指定しているため容易に変更できる(パラメータ値を修正した後コンパイルを行うことによって変更できる)。各パラメータの変数名は以下の通りであり、インクルードコモンのファイルに登録されている。
- ゲート総数 : YINGMX
 - コンポーネント総数 : YINCMX
 - ミニマルカットセット総数 : YMCSMX

なお、CUT-TDコードのソースプログラムはFORTRAN77(一部アセンブラ)で記述され、62個のサブルーチン(うち4個はアセンブラルーチン)から構成される。プログラムの大きさは約8000ステップで、ロードモジュールは約1.1メガバイト、実行時の主記憶領域は約3.1メガバイトである。

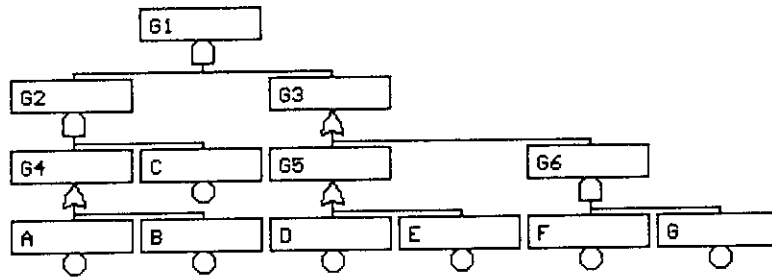


Fig. 3.1 簡単なフォールトツリーの例

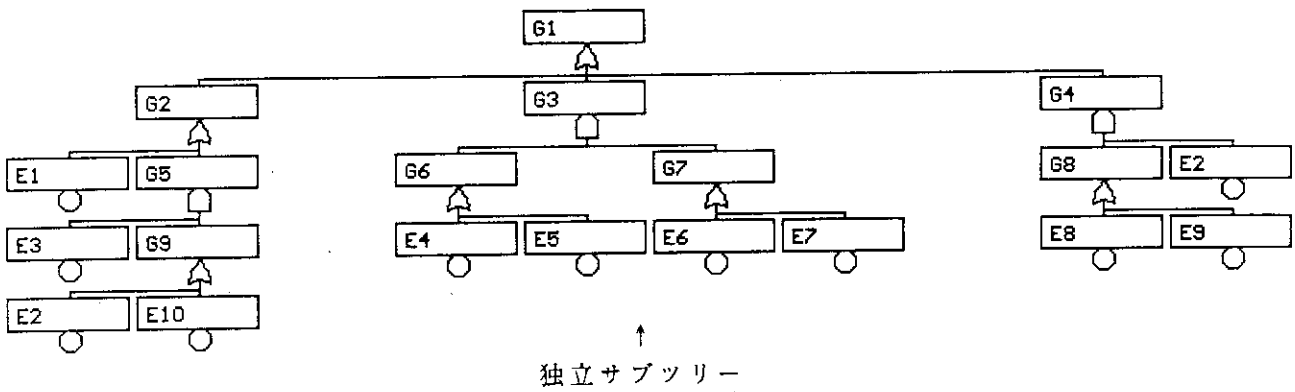


Fig. 3.2 独立サブツリーを含むフォールトツリーの例

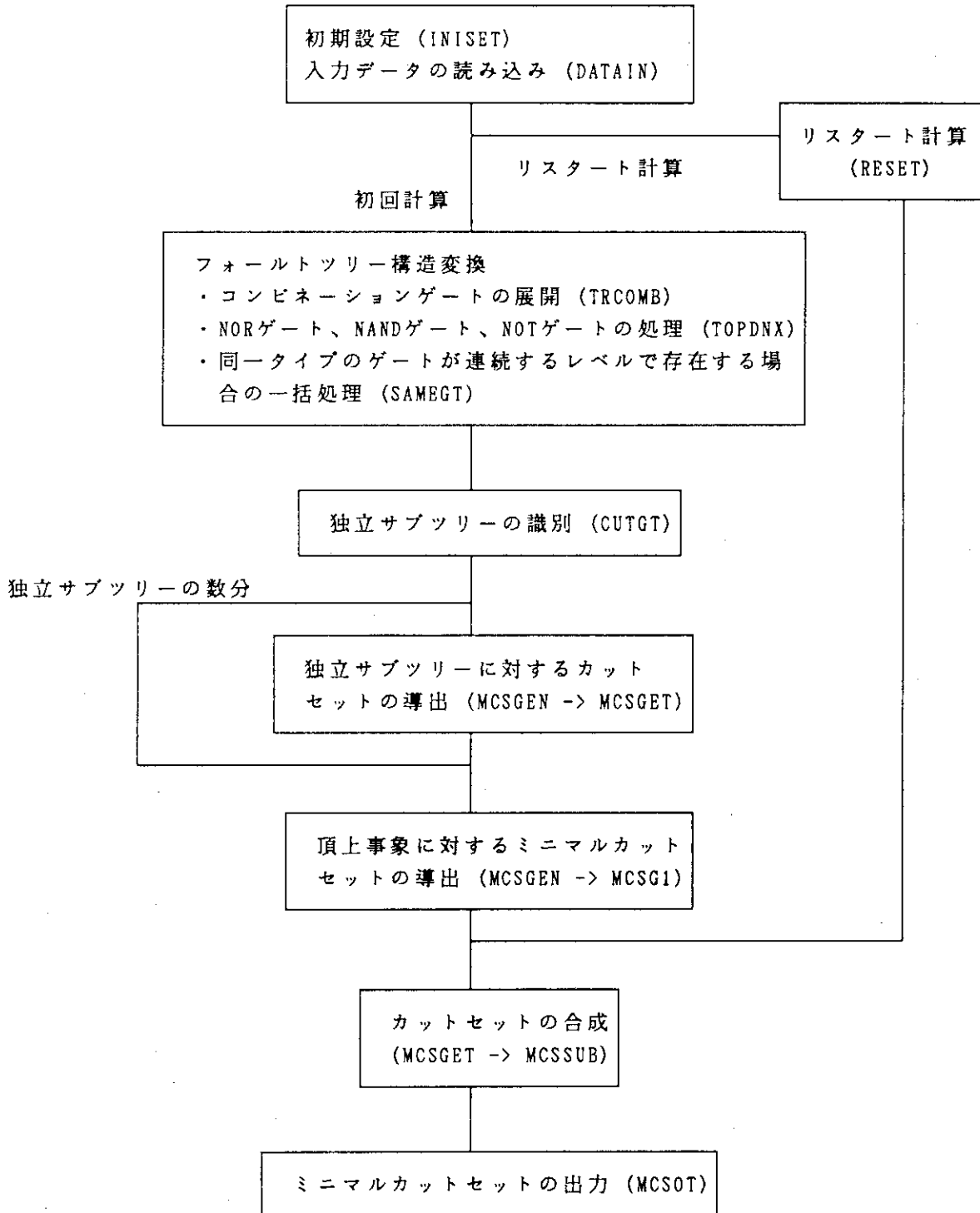


Fig. 3.3 CUT-TDコードにおける処理フロー概略

4. CUT-TDコードの入出力

4.1 入出力ファイル構成

CUT-TDコードの入出力ファイル構成をFig.4.1に、また、各ファイルの役割及び属性をTable 4.1に示す。これらの図表に示すように、CUT-TDコードは、3種類の入力データファイル（ゲート情報ファイル、コンポーネント情報ファイル、及び、計算制御データファイル）、2種類のリスタート用ファイル、及び、3種類の出力ファイル（ミニマルカットセット情報出力ファイル、計算結果出力ファイル、及び、フォールトツリー構造データ等入力情報のリスト出力用ファイル）を必要とする。なお、出力ファイルのうち、計算結果及び入力情報の出力先は、プリンタと設定するのが一般的な使用方法である。

4.2 入力データ

前節に述べたように、CUT-TDコードでは、以下の3種類の入力データファイルを必要とする。

(1) ゲート情報ファイル

このファイルはフォールトツリーの構造に関するデータを収録するためのものであり、Fig.4.2にデータフォーマットを示す。同図に示すように、フォールトツリーの各ゲートに対する情報（ゲートの入力条件と事象記述）は2レコードより構成される。CUT-TDコードでは、1レコード目だけを使用し、2レコード目は入力しなくても支障はない。なお、データフォーマットは、FTA-Jコードで作成されるゲート情報と同一のものである（即ち、FTA-JコードのNF03ファイルに相当する）。以下、各データ項目の意味と入力型式について記述する。

1レコード目 --- 各ゲートに対する入力条件

- ① GATEN(I) : 当該ゲートの名前 (A8、左詰め：1-8カラム)
- ② GATET(I) : 当該ゲートのタイプ (A4、左詰め：10-13カラム)
- ③ ING(I) : 当該ゲートの入力ゲート数 (I1、 $0 \leq \text{ING} \leq 8$: 14カラム)
- ④ INC(I) : 当該ゲートの入力コンポーネント数 (I1、 $0 \leq \text{INC} \leq 8$: 15カラム)
- ⑤ ICOM(I) : コンビネーションゲートの条件数 (I1、 $0 \leq \text{ICOM} \leq 8$: 16カラム)
- ⑥ GCNAM(I, 1)~GCNAM(I, 8) : 当該ゲートの入力ゲート名/コンポーネント名
(A8、左詰め、最大8個：17-24, 25-32, 33-40, 41-48, 49-56, 57-64, 65-72, 73-80カラム)

左からING(I)で指定した個数をゲートとみなし、その後にINC(I)で指定した個数をコンポーネントとみなす。従って、ING(I)+INC(I)個のゲート名/コンポーネント名が許される。

2レコード目 --- 各ゲートの事象記述 (コメント)

- ① DESCG(I) : ゲートに関する説明 (A54 : 5-58カラム)

ゲート情報は、上記2レコード1組として、フォールトツリー上に出現するゲートの数だけ与える必要がある。

(2) コンポーネント情報ファイル

このファイルは、解析対象とするフォールトツリーを構成するコンポーネントの生起確率及び事象説明に関するデータを収録するためのものであり、Fig. 4.3に示すように、各コンポーネントの情報は3レコードから構成される。ミニマルカットセットの導出及び頂上事象の発生確率の計算には1レコード目だけ使用するため、2レコード目と3レコード目は入力しなくても計算上支障はない。但し、2レコード目は不確実さ解析コードSPASM-Rの入力作成のためのデータであり、CUT-TDコードの計算結果を用いて不確実さ解析を行う場合には必要となる。また、3レコード目は、各コンポーネントに対する事象記述でありユーザにとってはいわばコメントとなる。なお、このファイルは、FTA-Jコードで作成されるコンポーネント情報ファイル（即ち、NF04ファイル）に相当するが、ファイルの容量及び作成上の労力を軽減するため、CUT-TDコードで必要とするデータだけを取り出しており、データフォーマットはFTA-Jとは異なっている。以下、各データ項目について記述する。

1レコード目 --- 点推定計算用のデータ

- ① COMPN(I) : 当該コンポーネントの名前 (A8、左詰め：1-8カラム)
- ② UNAV1(I) : 当該コンポーネントの生起確率 (E10.3 : 31-40カラム)

2レコード目 --- 不確実さ解析用のデータ

- ① MEAN(I) : 生起確率に対する正規分布の平均値 (E10.3 : 11-20カラム)
- ② STDEV(I) : 生起確率に対する正規分布の分散 (E10.3 : 21-30カラム)
- ③ MEDIAN(I) : 生起確率に対する対数正規分布の中央値 (E10.3 : 31-40カラム)
- ④ EF(I) : 生起確率に対する対数正規分布のエラーファクタ (E10.3 : 41-50カラム)
- ⑤ MIN(I) : 生起確率に対する対数一様分布の最小値 (E10.3 : 51-60カラム)
- ⑥ MAX(I) : 生起確率に対する対数一様分布の最大値 (E10.3 : 61-70カラム)
- ⑦ ALPHA(I) : 生起確率に対するガンマ分布の α 値 (E10.3 : 71-80カラム)
- ⑧ BETA(I) : 生起確率に対するガンマ分布の β 値 (E10.3 : 81-90カラム)

このレコードのデータは、CUT-TDコードのミニマルカットセット出力データの一部としてファイル出力される。

3レコード目 --- 各コンポーネントの事象記述 (コメント)

- ① DESC(C) : コンポーネントに関する説明 (A54 : 11-64カラム)

コンポーネント情報は、上記3レコード1組として、フォールトツリー上に出現するコンポーネントの数だけ与える必要がある。

(3) 計算制御用データファイル

計算制御用データは、ミニマルカットセット導出時の打ち切り条件や計算打ち切り時間等のオプションパラメータから構成される。Fig. 4.4にデータフォーマットを示す。

1レコード目 --- タイトルデータ (コメント)

- ① TITLE : 解析対象フォールトツリーのタイトル (A72 : 1-72カラム)

2レコード目 --- オプションパラメータ

- ① ISWSR : リスタート計算用スイッチ (I5 : 1-5カラム) --- デフォルト : 0

ISWSR=0 : 初回計算、ISWSR≠0 : リスタート計算

なお、下記のオプションパラメータのうちリスタート計算指定時には打ち切り時間のみ有効となる。

- ② LSEC : 計算打ち切り時間 (I5 : 6-10カラム)
--- デフォルト : 1800 (秒)、最小値 : 120 (秒)

- ③ MXJI: ミニマルカットセットの打切り次数 (I5: 11-15カラム)
--- デフォルト: 5、最大値: 8
- ④ PMIN: ミニマルカットセットの打切り確率 (F10.0: 16-25カラム)
--- デフォルト: 0.0
- ⑤ NEXG: 展開不要ゲートの数 (I5: 26-30カラム) --- デフォルト: 0
NEXG<0の時は、各ゲートごとにカットセットの数のみ計算する (この数は、巾等則及び吸収則を適用する前のカットセット総数である)。
- ⑥ MGAT: カットセットを出力するゲートの数 (I5: 31-35カラム)
--- デフォルト: 0
- ⑦ NTAB: MCSゲートの数 (I5: 36-40カラム) --- デフォルト: 0
- ⑧ IMPTNT: 生起確率、重要度の計算オプション (I5: 41-45カラム)
IMPTNT=0: ミニマルカットセットの導出のみ (次数順に出力)
=1: ミニマルカットセットの生起確率計算 (確率順に出力)
=2: ミニマルカットセットの重要度計算
(計算式: $P(MCS)_i / \sum P(MCS)_i$)
=3: ミニマルカットセットの重要度計算
(計算式: $P(MCS)_i /$ 下記⑨の入力値)
- ⑨ VINP: 上記⑧でIMPTNT=3とした時に分母になる確率 (F10.0: 46-55カラム)
--- デフォルト: 1.0
FTA-J (BAM) コード等で計算された頂上事象発生確率を入力する。

3 レコード目以下 --- 各種オプション指示フラグ等

- ① "NEXG" (固定: 1-4カラム): 展開不要ゲート指示フラグ
--- NEXG ≤ 0ならば不要
- ② NEXGN(I): 展開不要ゲート名 (A8, 2X: 1カラムからNEXG個指定)
--- NEXG ≤ 0ならば不要
なお、展開不要ゲートと指定されたゲートは疑似コンポーネントと見做され、その生起確率は1.0となる。
- ③ "MGAT" (固定: 1-4カラム): カットセットを出力するゲートの指示フラグ
--- MGAT ≤ 0ならば不要
- ④ GATENN(I), JOUT2(I): カットセットを出力するゲート名及びリスタート用スイッチ (A8, 1I, 1X: 1カラムからMGAT個指定)
--- MGAT ≤ 0ならば不要
JOUT2=0: 計算終了時間に達した時点でプログラムの実行を中止する。
≠1: 当該ゲートの展開処理が終了した時点でプログラムの実行を中止する。
- ⑤ "NTAB" (固定: 1-4カラム): MCSゲート指示フラグ --- NTAB ≤ 0ならば不要
- ⑥ ETAB(I): MCSゲート名 (A8: 1-8カラム) --- NTAB ≤ 0ならば不要
NTCS(I): 当該ゲートのMCS数 (I5: 11-15カラム) --- NTAB ≤ 0ならば不要
MCSゲートは、フォールトツリー上ではコンポーネントと見做されるため、コンポーネント情報ファイルにゲート名を登録しその生起確率を入力する必要がある。
- ⑦ JI(J): MCSの構成コンポーネント数 (I3: 1-3カラム)
--- $1 \leq JI \leq 8$, NTAB ≤ 0ならば不要
SHARP(J, K), EETAB(J, K): "NOT"型コンポーネント識別フラグ、MCS構成コンポーネント名 (A1, A8: 4カラムからJI個指定)
--- NTAB ≤ 0ならば不要

SHARP(J, K) = "#": "NOT"型コンポーネント
 = " ": 通常のコンポーネント

なお、⑥及び⑦は、NTAB個必要となる。

4.3 出力情報

Fig. 4.1に示したように、CUT-TDコードの出力ファイルには以下の3種類がある。

(1) ミニマルカットセット情報ファイル

CUT-TDコードで求めたミニマルカットセット及びそれを構成するコンポーネントに関する情報（各コンポーネントの生起確率に対する分布パラメータ）を収録したファイルであり、フォールトツリーの頂上事象発生確率に対する不確かさ解析コードSPASM-Rの入力データとなる。Fig. 4.5にデータフォーマットを示す。

ミニマルカットセットに関する情報

- ① NELM(I) : MCSの次数 ($14, 1 \leq n \leq 10$: 1-4カラム)
- ② SHARP(I, J)、CSTR(I, J) : "NOT"型コンポーネント識別フラグ、MCS構成コンポーネント名 (A1, A8 : 5カラムからNELM個指定)

SHARP(I, J) = "#": "NOT"型コンポーネント
 = " ": 通常のコンポーネント

①及び②は、ミニマルカットセットの数分（即ち、1個）出力される。

構成コンポーネントに関する情報

- ① IWRK(K) : MCS構成コンポーネントであることを示すフラグ (I4 : 1-4カラム)
 IWRK(K) = 0と出力される。
- ② CCMP(K) : MCS構成コンポーネント名 (A8、5-12カラム)
- ③ MEAN(K) : 生起確率に対する正規分布の平均値 (E10.3 : 13-22カラム)
- ④ STDEV(K) : 生起確率に対する正規分布の分散 (E10.3 : 23-32カラム)
- ⑤ MEDIAN(K) : 生起確率に対する対数正規分布の中央値 (E10.3 : 33-42カラム)
- ⑥ EF(K) : 生起確率に対する対数正規分布のエラーファクタ (E10.3 : 43-52カラム)
- ⑦ MIN(K) : 生起確率に対する対数一様分布の最小値 (E10.3 : 53-62カラム)
- ⑧ MAX(K) : 生起確率に対する対数一様分布の最大値 (E10.3 : 63-72カラム)
- ⑨ ALPHA(K) : 生起確率に対するガンマ分布の α 値 (E10.3 : 73-82カラム)
- ⑩ BETA(K) : 生起確率に対するガンマ分布の β 値 (E10.3 : 83-92カラム)

これらのデータは、ミニマルカットセットを構成するコンポーネント全て（K個）について出力される。

(2) 計算結果

CUT-TDコードの計算結果は以下の項目から成る。出力機番は6であり出力先として通常はプリンタを設定する。

- ・ 計算制御用オプションパラメータ (Fig. 4.6)
 ユーザ入力で指定したオプションパラメータが出力される。
- ・ 独立サブツリーに関する情報 (Fig. 4.7)
 解析対象としたフォールトツリーの中で独立な部分を識別し、該当するゲート名及びそれ以下のツリー構造が出力される。
- ・ ミニマルカットセットに関する情報 (Fig. 4.8)
 頂上事象に対するミニマルカットセットの生起確率（あるいは重要度）、次数及び構成コンポーネント名、頂上事象の発生確率が出力される。ミニマルカットセ

ットは、生起確率の大きさ順あるいは次数順のいずれかに従って出力される（確率の順番か次数順かはユーザによる入力指定で、計算制御用オプションパラメータの1つである重要度計算オプションIMPNTを"0"とした場合にのみ次数順になる）。また、ユーザ入力で指定したゲートについても、当該ゲートに対するミニマルカットセットが出力される。

(3) フォールトツリー構造データ等入力情報

補助的な出力として、解析対象としたフォールトツリーに関する情報が出力される。出力機番は7であり、出力先として通常プリンタを設定する。出力項目は以下の通りである。

- ・ フォールトツリー上に現れるコンポーネントの生起確率 (Fig. 4.9)
- ・ フォールトツリーの構造データ (Fig. 4.10)
- ・ ゲート-コンポーネントのクロスリファレンス情報 (Fig. 4.11)
- ・ ゲート-ゲートのクロスリファレンス情報 (Fig. 4.12)

4.4 エラーメッセージ

本節では、CUT-TDコードのエラーメッセージについてその原因及び対策を記述する。

"MGAT-CARD MISSING THEN STOP"

原因：計算制御用オプションパラメータ、カットセット出力ゲート指示フラグ"MGAT"が存在しない。

対策：入力データ（計算制御用オプションパラメータ）をチェックする。

"NTAB-CARD MISSING THEN STOP"

原因：計算制御用オプションパラメータ、MCSゲート指示フラグ"NTAB"が存在しない。

対策：入力データ（計算制御用オプションパラメータ）をチェックする。

"ETAB, NTCS-CARD MISSING THEN STOP"

原因：計算制御用オプションパラメータ、MCSゲート指示フラグ"MGAT"が存在するにも拘らずMCSゲート名(ETAB)あるいはMCSゲートの構成コンポーネント数(NTCS)が入力されていない。

対策：入力データ（計算制御用オプションパラメータ）をチェックする。

"NTCS MORE THAN INGMX THEN STOP"

原因：計算制御用オプションパラメータ、MCSゲートに対するミニマルカットセットの数として入力ゲートの最大値 (INGMX=500) を超えた値が入力されている。

対策：入力データ（計算制御用オプションパラメータ）をチェックする。

"INPUT MCS DATA MISSING THEN STOP"

原因：計算制御用オプションパラメータ、MCSゲートに対するミニマルカットセットの入力データに誤りがある（例、1つのミニマルカットセットの構成コンポーネント数が8より大きい）。

対策：入力データ（計算制御用オプションパラメータ）をチェックする。

"NEXG-CARD MISSING THEN STOP"

原因：計算制御用オプションパラメータ、展開不要ゲート指示フラグ"NEXG"が存在しない。

対策：入力データ（計算制御用オプションパラメータ）をチェックする。

"TOO MANY GATES (INPTRE)"

原因：解析対象とするフォールトツリーに存在するゲート数が最大値（500個）を超えた場合に出力される。

対策：対策としては以下の2通りが考えられる。

- ・解析対象とするフォールトツリーを縮小する。
- ・プログラム内のゲート数最大値を変更する（パラメータ文`YINGMX=500`を変更する）。なお、この場合には、プログラム実行時の主記憶領域を大きくする必要はある。

"TOO MANY COMPONENTS (INPTRE)"

原因：解析対象とするフォールトツリーに存在するコンポーネント数が最大値（500個）を超えた場合に出力される。

対策：対策としては以下の2通りが考えられる。

- ・解析対象とするフォールトツリーを縮小する。
- ・プログラム内のコンポーネント数最大値を変更する（パラメータ文`YINCMX=500`を変更する）。なお、この場合には、プログラム実行時の主記憶領域を大きくする必要はある。

"GATE TABLE OVERFLOW THEN STOP (TRCOMB)"

原因：コンビネーションゲートを展開した後のゲート総数がゲート数の最大値（500個）を超えた場合に出力される。

対策：対策としては以下の2通りが考えられる。

- ・解析対象とするフォールトツリーを縮小する。
- ・プログラム内のゲート数最大値を変更する（パラメータ文`YINGMX=500`を変更する）。なお、この場合には、プログラム実行時の主記憶領域を大きくする必要はある。

"COMPONENT TABLE OVERFLOW THEN STOP (TRNOT)"

原因：NANDゲート、NORゲート、NOTゲートを展開した後のコンポーネント総数がコンポーネント数の最大値（500個）を超えた場合に出力される。

対策：対策としては以下の2通りが考えられる。

- ・解析対象とするフォールトツリーを縮小する。
- ・プログラム内のコンポーネント数最大値を変更する（パラメータ文`YINCMX=500`を変更する）。なお、この場合には、プログラム実行時の主記憶領域を大きくする必要はある。

"GATE TABLE OVERFLOW THEN STOP (TRNOT)"

原因：NANDゲート、NORゲート、NOTゲートを展開した後のゲート総数がゲート数の最大値（500個）を超えた場合に出力される。

対策：対策としては以下の2通りが考えられる。

- ・解析対象とするフォールトツリーを縮小する。
- ・プログラム内のゲート数最大値を変更する（パラメータ文`YINGMX=500`を変更する）。なお、この場合には、プログラム実行時の主記憶領域を大きくする必要はある。

"TXREF xxxxxxxx NOT EXIST IN GATE TABLE THEN STOP"

原因：ゲートxxxxxxxが存在しない場合に出力される。

対策：解析対象フォールトツリーの構造データ（ゲート情報ファイル）をチェックする。

"TXREF xxxxxxxx NOT EXIST IN COMPONENT TABLE THEN STOP"

原因：コンポーネントxxxxxxxが存在しない場合に出力される。

対策：解析対象フォールトツリーのコンポーネントに関する情報（コンポーネント情報ファイル）をチェックする。

"ICCTAB OVERFLOW (CUTGT) THEN STOP"

原因：CUT-TDコードでは、解析対象とするフォールトツリーを独立サブツリーをはじめ幾つかの小規模ツリーに分割しているが、その分割したツリーの数が増大（2000個）を超えた場合にこのメッセージが出力される。

対策：対策としては以下の2通りが考えられる。

- ・ 解析対象とするフォールトツリーを縮小する。
- ・ プログラム内の分割ツリー数最大値を変更する（パラメータ文`YIPTMX=2000`を変更する）。

"MCS TABLE OVERFLOW (MCSGET)""MCS TABLE OVERFLOW (MCSSUB)"

原因：ミニマルカットセットを求めている途中の段階（サブルーチンMCSGETあるいはMCSSUB）でカットセットの総数が最大値（20000個）を超えた場合にこのメッセージが出力される。

対策：対策としては以下の3通りが考えられる。

- ・ 解析対象とするフォールトツリーを縮小する。
- ・ ミニマルカットセットを求めるための打ち切り条件を変更する（即ち、打ち切り確率を大きくするか、あるいは、打ち切り次数を小さくする）。
- ・ プログラム内のカットセット総数最大値を変更する（パラメータ文`YMCSMX=20000`を変更する）。なお、この場合には、プログラム実行時の主記憶領域を大きくする必要がある。

"STACK OVERFLOW THEN STOP (PUSH)""STACK OVERFLOW THEN STOP (TOPDWN)"

原因：ミニマルカットセットを求めている途中の段階（サブルーチンPUSHあるいはTOPDWN）でスタックエリアの大きさが最大値（2500）を超えた場合にこのメッセージが出力される。

対策：対策としては以下の2通りが考えられる。

- ・ 解析対象とするフォールトツリーを縮小する。
- ・ プログラム内のスタックエリア最大値を変更する（パラメータ文`YSTCK=2500`を変更する）。なお、この場合には、プログラム実行時の主記憶領域を大きくする必要がある。

Table 4.1 CUT-TDコードの構成ファイル一覧

機番	変数名	収録データ	ファイル属性(LRECL, BLKSIZE, RECFM)
FT05F001	INPT	計算制御パラメータ	LRECL=80, BLKSIZE=80n, RECFM=FB
FT06F001	IOLP	計算結果の出力	LRECL=135, BLKSIZE=143, RECFM=VBA
FT07F001	IPLP	入力データのリスト出力 クロスリファレンス情報	LRECL=135, BLKSIZE=143, RECFM=VBA
FT13F001	NF3	ゲート情報	LRECL=80, BLKSIZE=80n, RECFM=FB
FT14F001	NF4	コンポーネント情報	LRECL=90, BLKSIZE=90n, RECFM=FB
FT16F001	NFILE	ミニマルカットセット情報 (計算結果)	LRECL=100, BLKSIZE=100n, RECFM=FB
FT20F001	MFILE	ミニマルカットセット情報 (リスタート用)	LRECL=22000, BLKSIZE=22008, RECFM=VB
FT27F001	IRFILE	リスタート用制御情報	LRECL=22000, BLKSIZE=22008, RECFM=VB

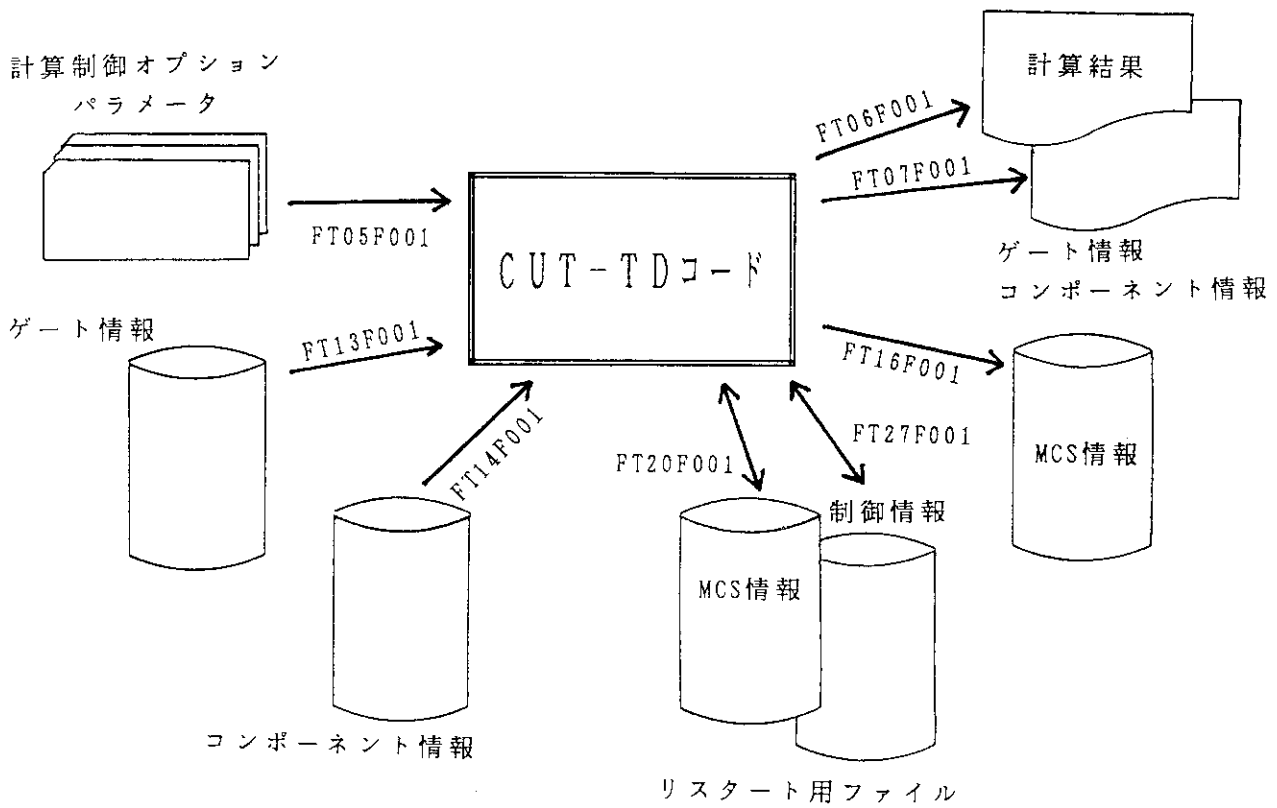


Fig. 4.1 CUT-TDコードの入出力ファイル構成

1		2		3		4		5		6		7		8					
1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0
① GATEN(I)		GATET		GCNAM(I,1)		GCNAM(I,2)		GCNAM(I,3)		GCNAM(I,4)		GCNAM(I,5)		GCNAM(I,6)		GCNAM(I,7)		GCNAM(I,8)	
③ INC(I)		⑤ ICOM(I)		④ INC(I)		⑦ DESCG(I): ゲートに関する説明 (A54)													
<p>2 レコード1組として、フォールトツリー上に出現するゲートの数だけ与える。</p> <p>① GATEN(I): ゲート名 (A8) ② GATET(I): ゲートタイプ (A4) ③ INC(I): 入力ゲート数 ($11: 0 \leq n \leq 8$) ④ INC(I): 入力コンポーネント数 ($11: 0 \leq n \leq 8$) ⑤ ICOM(I): コンビネーションゲートの条件数 ($11: 0 \leq n \leq 8$) ⑥ GCNAM(I,1)~GCNAM(I,8): 入力ゲート名/コンポーネント名 (A8) 左からINC(I)で指定した個数をゲートとみなし、その後にINC(I)で指定した個数をコンポーネントとみなす。</p>																			
1		2		3		4		5		6		7		8					
1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0

共通E173

*73~75: DATAID/76~80: SEQUENTIAL NUMBER

Fig. 4.2 ゲート情報ファイル・データフォーマット

1 / 0 #	① COMPN (1) コンポーネント名 (A8)	② UNAVI (1) 発生確率 (E10.3)					
1 / 0 #			正規分布パラメータ	対数正規分布パラメータ	対数一様分布パラメータ	ガンマ分布パラメータ	
	③ MEAN (1) 平均値 (E10.3)	④ STDEV (1) 分散*1.64 (E10.3)	⑤ MEDIAN (1) 中央値 (E10.3)	⑥ EF (1) E _F -7777 (E10.3)	⑦ MIN (1) 最小値 (E10.3)	⑧ ALPHA (1) α 値 (E10.3)	⑨ BETA (1) β 値 (E10.3)
1 / 0 #			⑩ BESSC (1) : コンポーネントに関する説明 (A54)				
1 / 0 #	3 レコード1 組として、フォールトツリー上に出現するコンポーネントの数だけ与える。						

Fig. 4.3 コンポーネント情報ファイル・データフォーマット

1	2	3	4	5	6	7	*	8
1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8
9	0	1	2	3	4	5	6	7
8	9	0	1	2	3	4	5	6
7	8	9	0	1	2	3	4	5
6	7	8	9	0	1	2	3	4
5	6	7	8	9	0	1	2	3
4	5	6	7	8	9	0	1	2
3	4	5	6	7	8	9	0	1
2	3	4	5	6	7	8	9	0
1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8
9	0	1	2	3	4	5	6	7
8	9	0	1	2	3	4	5	6
7	8	9	0	1	2	3	4	5
6	7	8	9	0	1	2	3	4
5	6	7	8	9	0	1	2	3
4	5	6	7	8	9	0	1	2
3	4	5	6	7	8	9	0	1
2	3	4	5	6	7	8	9	0
1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8

TITLE: 解析対象フォールトツリーのタイトル (A72)

①	②	③	④	⑤	⑥	⑦	⑧	⑨
ISWSR	LSEC	MXJI	PMIN	NEXG	MGAT	NTAB	IMPTNT	VINP

① ISWSR: リスタート計算用スイッチ (15、デフォルト: 0)
 ISWSR=0: 初回計算、ISWSR≠0: リスタート計算
 リスタート計算指定時には打ち切り時間のみ有効となる。

② LSEC: 計算打ち切り時間 (15、デフォルト: 1800秒、最小値: 120秒)

③ MXJI: ミニマルカットセットの打ち切り回数 (15、デフォルト: 5)

④ PMIN: ミニマルカットセットの打ち切り確率 (F10.0、デフォルト: 0.0)

⑤ NEXG: 展開不要ゲートの数 (15、デフォルト: 0)
 NEXG<0の時、各ゲートごとにカットセットの数のみ計算する。

⑥ MGAT: カットセットを出力する中間ゲートの数 (15、デフォルト: 0)

⑦ NTAB: MCSゲートの数 (15、デフォルト: 0)

⑧ IMPTNT: 発生確率、重要度の計算オプション (15)
 IMPTNT=0: ミニマルカットセットの算出のみ (次数順に出力)
 =1: ミニマルカットセットの生起確率計算 (確率順に出力)
 =2: ミニマルカットセットの重要度計算 (計算式: P(MCS)i / Σ P(MCS)i)
 =3: ミニマルカットセットの重要度計算 (計算式: P(MCS)i / 下記⑨の人力値)

⑨ VINP: 上記⑧でIMPTNT=3とした時に分母になる確率 (F10.0、デフォルト: 1.0)

Fig. 4.4 計算制御データファイル・データフォーマット (その1)

1										2										3										4										5										6										7										*										8																																																																																																																																											
1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0																																																																																																																																		
NEXG										展開不要ゲート指示フラグ (固定)																																																																																																																																																																																																																	
NEXGN(1)																																																																																																																																																																																																																											
MGAT										カットセット出力ゲートの指示フラグ (固定)																																																																																																																																																																																																																	
GATENN(1)																																																																																																																																																																																																																											

*7.3~7.5: DATA/D/76~80: SEQUENTIAL NUMBER

Fig. 4.4 計算制御データファイル・フォーマット (その2)

1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	2 1 2 3 4 5 6 7 8 9 0	3 0 1 2 3 4 5 6 7 8 9 0	4 0 1 2 3 4 5 6 7 8 9 0	5 0 1 2 3 4 5 6 7 8 9 0	6 0 1 2 3 4 5 6 7 8 9 0	7 0 1 2 3 4 5 6 7 8 9 0	8 0 1 2 3 4 5 6 7 8 9 0	
NTAB									
MCSゲート指示フラグ (固定)									
ETAB(I)									
NTCS(I)									
JI(J)									
EETAB(J, K)									
SHARP(J, K)									
NTCS(I) 個必要									
ETAB(I) : MCSゲート名 (A8)									
NTCS(I) : 当該ゲートのMCS数 (15)									
JI(J) : MCSの構成コンポーネント数 (13: 1 ≤ JI ≤ 8)									
SHARP(J, K) : "NOT"型コンポーネント識別フラグ (A1)									
SHARP(J, K) = "#": "NOT"型コンポーネント =" ": 通常のコンポーネント									
EETAB(J, K) : MCS構成コンポーネント名 (A8: JI個指定)									
NTAB ≤ 0 ならば 不要									
NTAB組必要									

*73-75: DATAID:76-80: SEQUENTIAL NUMBER

Fig. 4.4 計算制御データファイル・データフォーマット (その3)

①		② MCSを構成するコンポーネント名 (次数の分)											
MELM (1)		CSTR(1,1) (A8)	CSTR(1,2) (A8)	CSTR(1,3) (A8)	CSTR(1,4) (A8)	CSTR(1,5) (A8)	CSTR(1,6) (A8)	CSTR(1,7) (A8)	CSTR(1,8) (A8)	CSTR(1,9) (A8)	CSTR(1,10) (A8)		
SHARP(1,1) : "NOT"型コンポーネント識別フラグ (A1) SHARP(1,1) = "#": "NOT"型コンポーネント = ".": 通常のコンポーネント													
①		正規分布パラメータ			対数正規分布パラメータ			対数二様分布パラメータ			ガンマ分布パラメータ		
IWRK (K)	CCMP (K) コンポーネント名 (A8)	③ MEAN (K) 平均値 (E10.3)	④ STDEV (K) 分散*1.64 (E10.3)	⑤ MEDIAN (K) 中央値 (E10.3)	⑥ BF (K) 17-7777 (E10.3)	⑦ MIN (K) 最小値 (E10.3)	⑧ MAX (K) 最大値 (E10.3)	⑨ ALPHA (K) α 値 (E10.3)	⑩ BETA (K) β 値 (E10.3)				

Fig. 4.5 ミニマルカセット情報出力ファイル・データフォーマット

***** CUT-TD VER. 1.0 -- * ** *** TEST CASE OF SETS EXAMPLE *** **

> INPUT PARAMETERS

```

* RESTART SWITCH                      NO
* CUT OFF CPU TIME FOR RESTART        3600 (SEC)
* MAXIMUM CUT OFF ORDER OF MCS        5
* MINIMUM CUT OFF PROBABILITY          0.0
* CALCULATE UPPER BOUND OF MCS# ONLY   NO
* NUMBER OF GATE TO BE PSEUDO COMPONENT 0
* NUMBER OF GATE FOR INTERMEDIATE RESULT 0
* NUMBER OF MCS TYPE GATE              0
* MCS IMPORTANCE CALCULATION OPTION     1
    
```

```

WHERE 0 -- NO IMPORTANCE (ORDERED BY NUMBER)
      1 -- MCS-UNAV
      2 -- MCS-UNAV / TOTAL-UNAV
      3 -- MCS-UNAV / INPUT-VALUE
      * INPUT VALUE = 0.5000E+00
    
```

Fig. 4.6 計算制御オプションパラメータの印刷出力 (例)

***** CUT-TD VER. 1.0 -- * ** *** TEST CASE OF SETS EXAMPLE *** **

*** SUB TREE LIST ***

```

(G1      ) INDEPENDENT      # OF CHILD MODULE = 2      LEVEL = 3
  G1      OR      3 0  G2      G3      G4
  G2      OR      1 1  G5      E1
  G5      AND     1 1  G9      E3
  G9      OR      0 2  E2      E10
  G3      AND     2 0  G6      G7
  G4      AND     1 1  G8      E2
  G8      OR      0 2  E8      E9
    
```

```

(G3      ) INDEPENDENT      # OF CHILD MODULE = 0      LEVEL = 1
  G3      AND     2 0  G6      G7
  G6      OR      0 2  E4      E5
  G7      OR      0 2  E6      E7
    
```

```

(G8      ) INDEPENDENT      # OF CHILD MODULE = 0      LEVEL = 0
  G8      OR      0 2  E8      E9
    
```

Fig. 4.7 独立サブツリーに関する情報の印刷出力 (例)

```

***** CUT-TD VER. 1.0 -- * ** *** TEST CASE OF SETS EXAMPLE *** ** *
CUT SETS FOR GATE G1          IMPORTANCE = 1    ORDERED BY PROBABILITY    TOTAL = 0.30D-02
1  0.30E-02                    E1
2  0.30E-05                    E2
3  0.10E-05                    E4
4  0.30E-06                    E5
5  0.30E-06                    E8
6  0.10E-06                    E2
7  0.10E-06                    E4
8  0.30E-07                    E5
9  0.10E-07                    E10

```

Fig. 4.8 ミニマルカットセットの印刷出力 (例)

```

***** CUT-TD VER. 1.0 -- * ** *** TEST CASE OF SETS EXAMPLE *** ** *
> COMPONENT LIST
COMP.  UNAV.  COMP.  UNAV.  COMP.  UNAV.  COMP.  UNAV.  COMP.  UNAV.
E7     0.100E-03  E6     0.100E-02  E4     0.100E-02  E5     0.300E-03  E1     0.300E-02
E3     0.100E-03  E2     0.100E-02  E10    0.100E-03  E8     0.300E-03  E9     0.300E-02

```

Fig. 4.9 コンポーネント生起確率の印刷出力 (例)

***** CUT-TD VER. 1.0 -- * ** *** TEST CASE OF SETS EXAMPLE *** ** *

> INPUT FAULT TREE DESCRIPTION

- (1) GATE NUMBER
- (2) GATE NAME
- (3) GATE TYPE
- (4) NUMBER OF GATE INPUT
- (5) NUMBER OF COMPONENT INPUT
- (6) NUMBER OF EVENTS IN COM GATE TO BE CONSIDERED AT ONE TIME
- (7)-(14) NAMES OF INPUT

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)
1	G1	OR	3	0	0	G2	G3	G4					
2	G2	OR	1	1	0	G5	E1						
3	G3	AND	2	0	0	G6	G7						
4	G4	AND	1	1	0	G8	E2						
5	G5	AND	1	1	0	G9	E3						
6	G6	OR	0	2	0	E4	E5						
7	G7	OR	0	2	0	E6	E7						
8	G8	OR	0	2	0	E8	E9						
9	G9	OR	0	2	0	E2	E10						

Fig. 4.10 フォールトツリー構造データの印刷出力 (例)

***** CUT-TD VER. 1.0 -- * ** *** TEST CASE OF SETS EXAMPLE *** ** *

> CROSS REFERENCE LIST

COMP NUMBER	COMP NAME	GATES INPUT TO
1	E7	G7
2	E6	G7
3	E4	G6
4	E5	G6
5	E1	G2
6	E3	G5
7	E2	G9
8	E10	G4
9	E8	G8
10	E9	G8

Fig. 4.11 ゲートコンポーネントのクロスリファレンス情報印刷出力 (例)

***** CUT-TD VER. 1.0 -- * ** *** TEST CASE OF SETS EXAMPLE *** ** *

> CROSS REFERENCE LIST

GATE NUMBER	GATE NAME	GATES INPUT TO
1	G1	
2	G2	G1
3	G3	G1
4	G4	G1
5	G5	G2
6	G6	G3
7	G7	G3
8	G8	G4
9	G9	G5

Fig. 4.12 ゲート---ゲートのクロスリファレンス情報印刷出力 (例)

5. サンプル計算

CUT-TDコードを用いて2種類のサンプル計算を行った。ひとつは、同一のフォールトツリーを対象にCUT-TDコードとFTA-Jコードを用いて計算を行い両コードの計算時間（CPUタイム）を比較するものであり、もうひとつは、CUT-TDコードの有効に利用するためのオプションを用いて計算を行い性能を確認するものである。本章では、これら2種類のサンプル計算について記述する。なお、以下に示すCPUタイムは原研の大型計算機FACOM M-780での計算実行時間である。

5.1 FTA-Jコードとの比較

CUT-TDコード及びFTA-Jコードを用いて、Fig. 5.1及びFig. 5.2に示す2つのフォールトツリー（フォールトツリー1及び2）各々を対象に、同一の計算条件で（打切り確率及び打切り次数を合わせて）ミニマルカットセットを求め計算時間の比較を行った。

(1) フォールトツリー1に対するサンプル計算

フォールトツリー1は、51個のゲート及び62個のコンポーネントから構成される。CUT-TDコード及びFTA-Jコード共に、ミニマルカットセットを求める際の打切り確率及び打切り次数は以下のように設定した。

- ・打切り確率：1.0E-12
- ・打切り次数：4次

両コードとも、計算結果として得られたミニマルカットセットの総数は270個であったが、計算に要したCPUタイムを比較すると、CUT-TDコードが1.5秒であったのに対しFTA-Jコードでは約30秒であった。

(2) フォールトツリー2に対するサンプル計算

フォールトツリー2は、SETS⁽⁸⁾のマニュアルに記載されているもので、40個のゲート及び34個のコンポーネントから構成される。CUT-TDコード及びFTA-Jコード共に、ミニマルカットセットを求める際の打切り確率及び打切り次数は以下のように設定した。

- ・打切り確率：0.0
- ・打切り次数：10次

両コードとも、計算結果として得られたミニマルカットセットの総数は994個であり、計算に要したCPUタイムは約30秒であった。

5.2 CUT-TDコードの性能確認

CUT-TDコードの性能を確認するために、Fig. 5.2のフォールトツリーを対象として、オプションを変えて計算を行った。以下、CUT-TDコードのオプションを変えた場合（ケース1～4）について記述する。

ケース1：ゲートのカットセット出力

Fig. 5.2のフォールトツリー上に存在するゲートG13とG32に対してカットセットを出力させた結果をFig. 5.3に示す。ゲートに対するカットセットは次数順に出力される。

ケース2：MCSゲートの指定

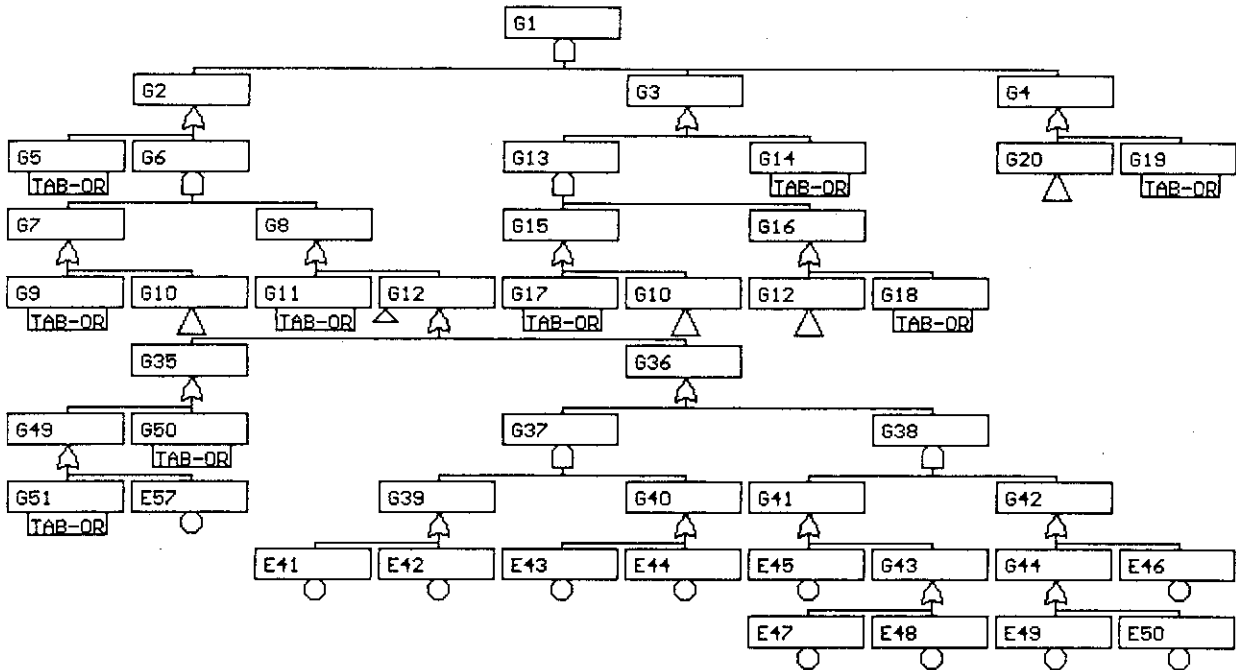
Fig. 5.2のフォールトツリーにおいて、G13とG39をMCSゲートと指定して計算を行った場合、頂上事象に対して994個のミニマルカットセットを求めるのに要したCPUタイムは約10秒程度となり計算時間が短縮される。

ケース3：展開不要ゲートの指定

Fig. 5.2のフォールトツリーにおいて、G13とG39を展開不要ゲートと指定して計算を行った場合、頂上事象に対して528個のミニマルカットセットが同定された。このオプションを指定して計算した場合には、展開不要ゲートの生起確率を1.0と自動的に設定するため、頂上事象の発生確率は正確なものとはならない場合があることに注意されたい。

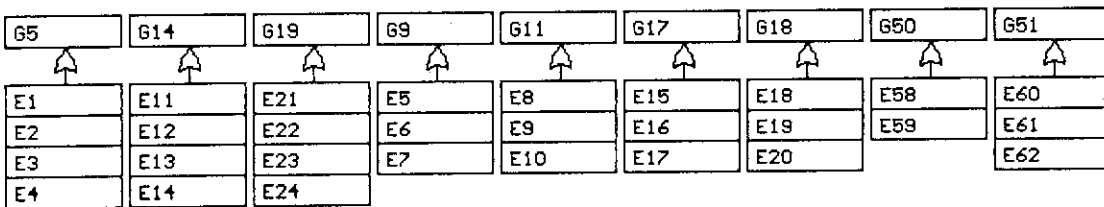
ケース4：ミニマルカットセットの重要度計算

頂上事象に対するミニマルカットセット各々の重要度を計算するオプションを指定して得た結果をFig. 5.4に示す。この例では、各ミニマルカットセットの重要度として、それぞれの生起確率を頂上事象の発生確率で割った値が示されている。



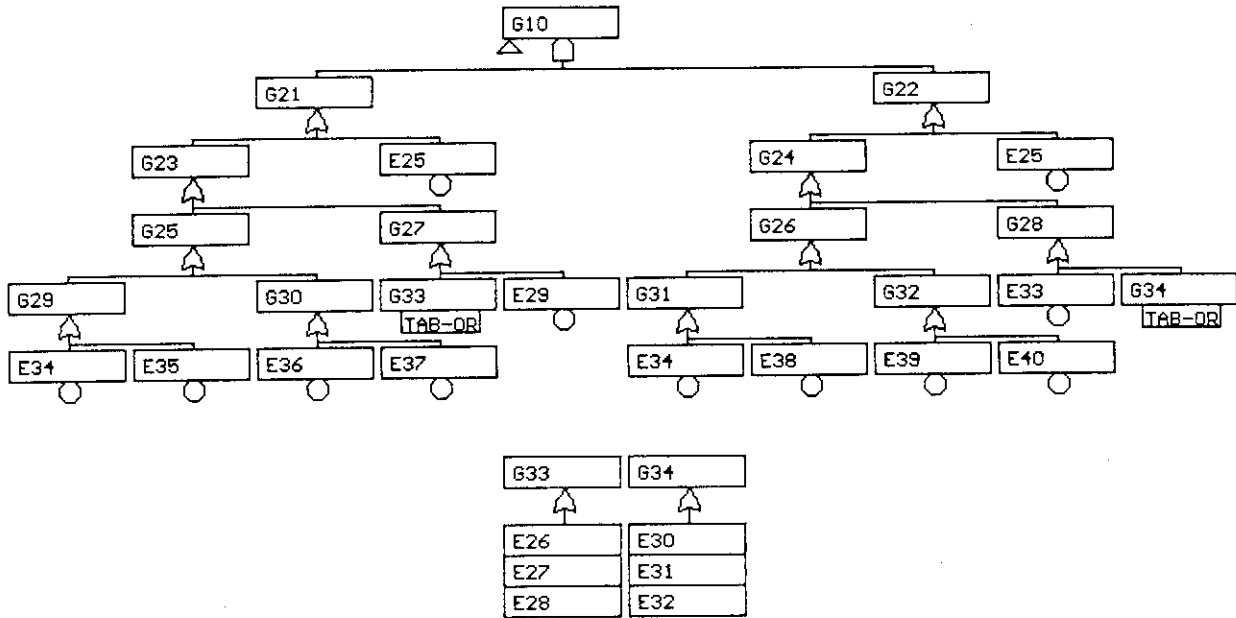
PAGE COMMENT:
FT COMMENT: CUT-TD Fig.5.1 Sample FT

P.1



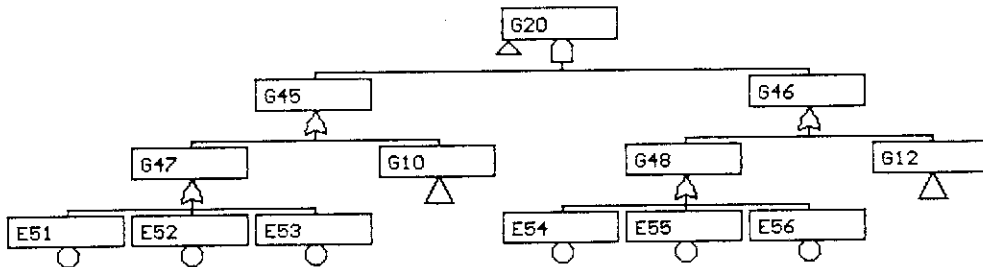
PAGE COMMENT:
FT COMMENT: CUT-TD Fig.5.1 Sample FT

Fig. 5.1 サンプル問題：フォールトツリー1 (その1)



PAGE COMMENT: G10
 FT COMMENT: CUT-TD Fig.5.1 Sample FT

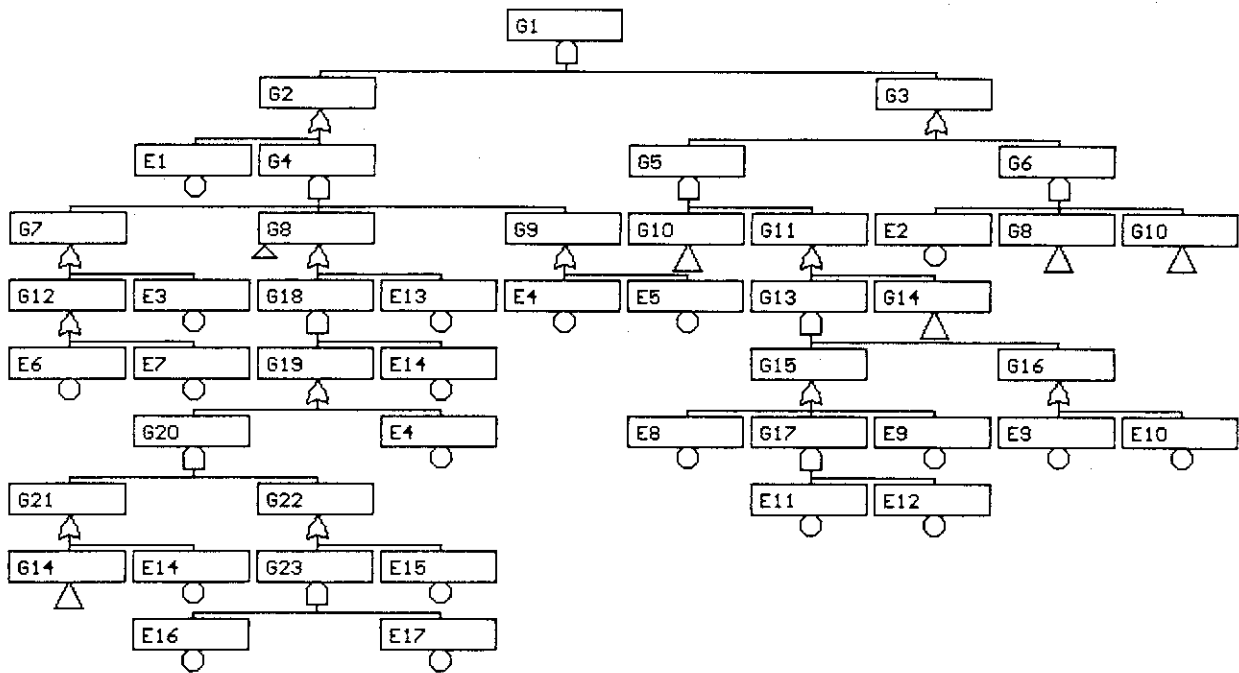
P.2



PAGE COMMENT: G20
 FT COMMENT: CUT-TD Fig.5.1 Sample FT

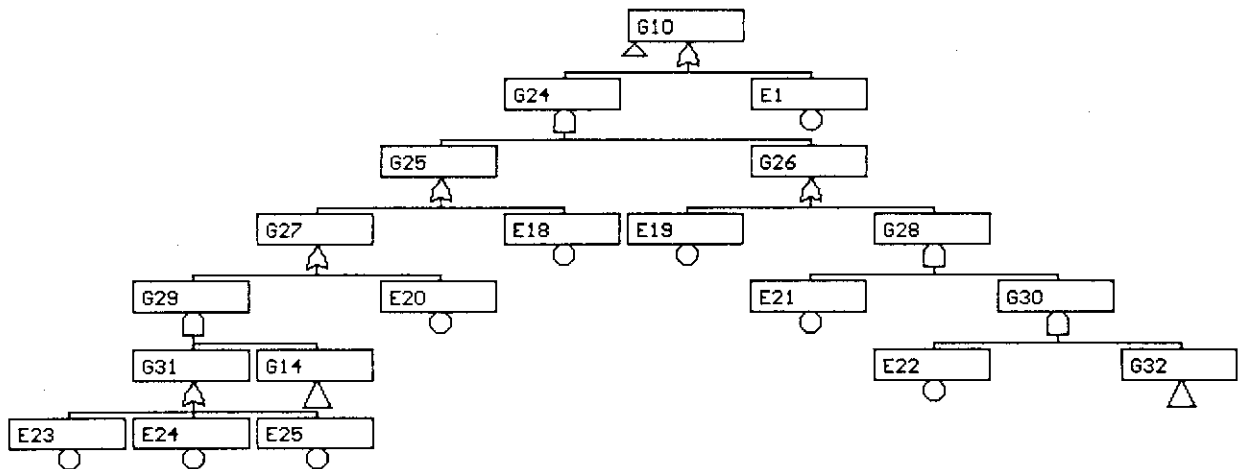
P.3

Fig. 5.1 サンプル問題：フォールトツリー 1 (その 2)



PAGE COMMENT:
FT COMMENT: CUT-TD Fig.5.2 Sample FT

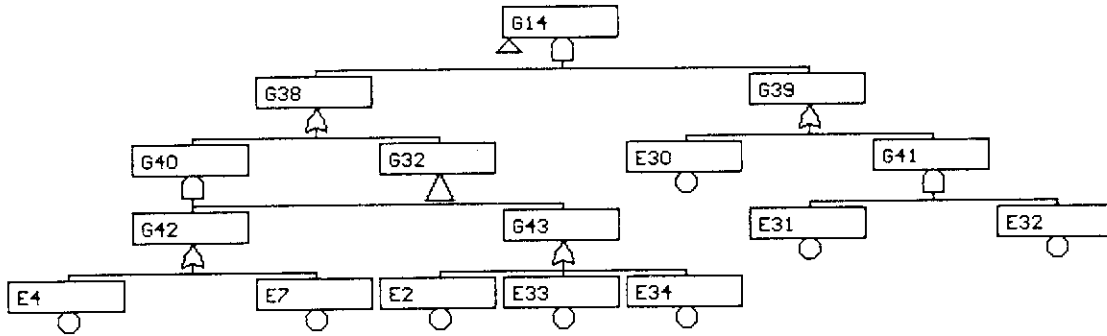
P.1



PAGE COMMENT: G10
FT COMMENT: CUT-TD Fig.5.2 Sample FT

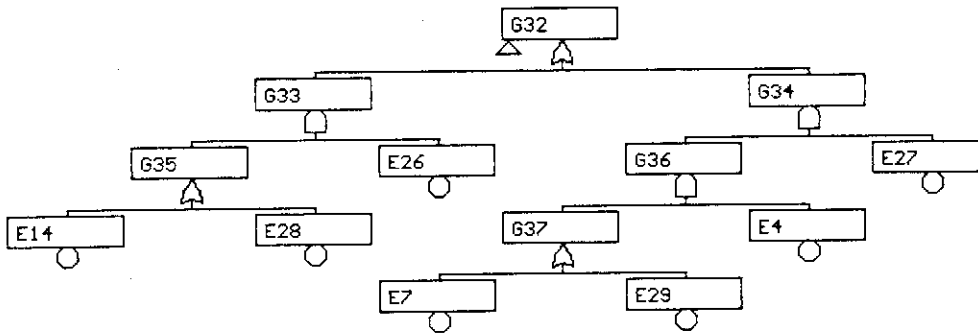
P.2

Fig. 5.2 サンプル問題：フォールトツリー 2 (その1)



PAGE COMMENT: G14
 FT COMMENT: CUT-TD Fig.5.2 Sample FT

P.3



PAGE COMMENT: G32
 FT COMMENT: CUT-TD Fig.5.2 Sample FT

P.4

Fig. 5.2 サンプル問題：フォールトツリー2（その2）

```

***** CUT-TD VER. 1.0 -- * * * * * TEST CASE OF SETS EXAMPLE * * * * *
CUT SETS FOR GATE G13                                     ORDERED BY MCS ORDER
1  0.10E-02  1  E9
2  0.10E-05  2  E8      E10
3  0.10E-08  3  E11     E12      E10

***** CUT-TD VER. 1.0 -- * * * * * TEST CASE OF SETS EXAMPLE * * * * *
CUT SETS FOR GATE G32                                     ORDERED BY MCS ORDER
1  0.10E-05  2  E14      E26
2  0.10E-05  2  E26     E28
3  0.10E-08  3  E7      E4      E27
4  0.10E-08  3  E4      E27     E29
    
```

Fig. 5.3 中間ゲートのカットセット出力結果

***** CUT-TD VER. 1.0 -- * ** *** TEST OF SAMPLE RUN SETS EXAMPLE : IMPORTANCE * * *

CUT SETS	FOR GATE	G1	IMPORTANCE = 2	ORDERED BY PROBABILITY	TOTAL = 0.100-05
1	0.10E+01	2	E1	E9	
2	0.10E-02	3	E1	E8	E20
3	0.10E-02	3	E1	E2	E13
4	0.10E-05	4	E1	E7	E2
5	0.10E-05	4	E1	E14	E30
6	0.10E-05	4	E1	E4	E26
7	0.10E-05	4	E1	E7	E30
8	0.10E-05	4	E1	E7	E34
9	0.10E-05	4	E1	E30	E28
10	0.10E-05	4	E1	E4	E30
11	0.10E-05	4	E1	E4	E33
12	0.10E-05	4	E1	E2	E34
13	0.10E-05	4	E1	E2	E15
14	0.10E-05	4	E1	E4	E14
15	0.10E-08	5	E1	E11	E10
16	0.10E-08	5	E1	E7	E4
17	0.10E-08	5	E1	E4	E30
18	0.10E-08	5	E1	E14	E27
19	0.10E-08	5	E1	E2	E31
20	0.10E-08	5	E1	E7	E16
21	0.10E-08	5	E1	E4	E32
22	0.10E-08	5	E1	E7	E31
23	0.10E-08	5	E1	E31	E32
24	0.10E-08	5	E1	E4	E28
25	0.10E-08	5	E1	E7	E32
26	0.10E-11	6	E6	E4	E31
27	0.10E-11	6	E6	E5	E32
28	0.10E-11	6	E3	E4	E19
29	0.10E-11	6	E3	E5	E19
30	0.10E-11	6	E7	E4	E20
31	0.10E-11	6	E6	E4	E20
32	0.10E-11	6	E1	E4	E29
33	0.10E-11	6	E7	E4	E18
34	0.10E-11	6	E7	E5	E18
35	0.10E-11	6	E6	E4	E19
36	0.10E-11	6	E6	E5	E18
37	0.10E-11	6	E3	E4	E20
38	0.10E-11	6	E3	E4	E19
39	0.10E-11	6	E3	E5	E18
40	0.10E-11	6	E7	E4	E18
41	0.10E-11	6	E6	E4	E19
42	0.10E-11	6	E1	E7	E27
43	0.10E-11	6	E7	E4	E20
44	0.10E-11	6	E7	E5	E20
45	0.10E-11	6	E6	E4	E20
46	0.10E-11	6	E6	E5	E20
47	0.10E-11	6	E3	E4	E20
48	0.10E-11	6	E3	E5	E20
49	0.10E-11	6	E7	E4	E20
50	0.10E-11	6	E6	E4	E20

Fig. 5.4 ミニマルカットセットの重要度計算結果

6. おわりに

本報告書は、フォールトツリー解析計算コードCUT-TDの使用手引としてをまとめたものである。CUT-TDコードの機能は、以下の通りである。

- ①フォールトツリーの頂上事象に対して、ミニマルカットセットを求める。
- ②フォールトツリー上の素事象（コンポーネント）の生起確率を用いて、各ミニマルカットセットの生起確率を計算すると共に、フォールトツリーの頂上事象の発生確率を計算する。
- ③不確実さ解析コードSPASM-Rの入力データファイルを作成する。
CUT-TDコードでは、ミニマルカットセットを求めるための計算効率を向上させるために次のような処理方法が用いられている。
 - (1) 同一タイプのゲート（"OR"ゲートあるいは"AND"ゲート）が連続したレベルで存在する場合には、その繋がり全体を1つのゲートにまとめ、当該ゲートに対するカットセットを求める。
 - (2) 独立サブツリーを自動的に認識し、各独立サブツリーごとにミニマルカットセットを求める。これは、ブール演算則（吸収則や巾等則）の処理に要する計算時間の軽減に繋がる。
 - (3) 既にミニマルカットセットが求められている事象をゲートとして入力することができ、これによって、複数のミニマルカットセット情報を結合して頂上事象に対するミニマルカットセットを求める。
 - (4) ミニマルカットセットを導出する段階で、次数による打ち切り、及び、生起確率による打ち切りが指定できる。これによって、頂上事象の発生確率に殆ど寄与しないようなミニマルカットセットを求めずに計算を実行することができる。
 - (5) ミニマルカットセットを求める段階で、展開を必要としないゲートを指定することができ、ゲートをコンポーネントと同等に扱う機能を有する。この機能は、解析者にとって不必要と思われるミニマルカットセットを求めることなく効率的に計算を行うことを目的とする。
 - (6) 計算時間（CPUタイム）をプログラム内部で管理しており、計算時間の打ち切り（CPUタイムオーバー）によって計算が異常終了するのを防ぎ、その時点までの途中結果をファイル出力する。さらに、この途中結果を用いて再実行することができる（リスタート機能）。

参考文献

- (1) 渡辺憲夫、樋口澄則、：“イベントツリー解析支援プログラムETAPの使用手引”，
JAERI-M 90-193, (1990)
- (2) 石神努他、：“フォールトツリー解析コードシステム：FTA-J使用手引”，
JAERI-M 83-169, (1983)
- (3) 及川哲邦他、：“機器故障率データベース，RECORDに登録された故障率集”
JAERI-M 89-044, (1989)
- (4) P.Chatterjee, :“Fault Tree Analysis : Reliability Theory and Systems
Safety Analysis”, ORC 74-34, Operations Research Center, University of
California, Berkeley, (1974)
- (5) R.C.Erdmann, et al., :“WAMCUT, A Computer Code for Fault Tree Evaluation”,
EPRI NP-803, (1978)
- (6) P.K.Pande, et al., :“Computerized Fault Tree Analysis : TREEL and MICSUP”,
ORC 75-3, Operations Research Center, University of California,
Berkeley, (1975)
- (7) J.B.Fussel and W.E.Vesely, :“A New Methodology for Obtaining Cut Sets”,
American Nuclear Society Transactions, Vol. 15, No. 1, pp. 262, (1972)
- (8) R.B.Worrell and D.W.Stack, :“A SETS User's Manual for the Fault Tree
Analyst”, NUREG/CR-0465, SAND77-2051, (1978)
- (9) B.Putney, et al., :“WAMCUT-II, A Fault Tree Evaluation Program”,
EPRI NP-2421, (1982)

付録 CUT-TDコードの構成

- (1) プログラムの構造
- (2) 主要サブルーチンの機能
- (3) 実行制御文

(1) プログラムの構造

```

MAIN  ----INISSET
      +---DATAIN----XDATE
      I              +---INPTRE
      I              +---TRAGC
      I              +---MKMCSF----TXREF3
      I              +---TXREF2
      I              +---*I1ASET
      I              +---CUTSET----*I1ATST
      I              +---MCSAPD
      +---RESET
      +---MCSNUM----COMBO
      +---TXREF ----TXREF1
      I              +---TXREF2
      I              +---TXREF4
      I              +---TXREF3
      +---TXROUT
      +---TRCOMB----COMBO
      +---TOPDNX----TRANS ----TRNOT
      I              +---LINK
      I              +---POP
      I              +---PUSH
      +---SAMEGT----REPGT
      +---CUTGT ----INDCHK----TOPDWN----LINK
      I              I              +---LCOMP
      I              I              +---POP
      I              I              +---PUSH
      I              +---XCLOCK
      I              +---RFWRT
      I              +---STOUT ----TOPDW2----LINK
      I              +---LCOMP
      I              +---POP
      I              +---LINK2
      I              +---PUSH
      +---MCSGEN----*I1ASET
      I              +---TENKAI----TOPDWN----LINK
      I              I              +---LCOMP
      I              I              +---POP
      I              I              +---PUSH
      I              I              +---BITH4 ----*I1ASET
      I              I              +---BITH3 ----*I1ATST
      I              +---MCSGET----MCSG1 ----BITHND----*I1ASET
      I              I              I              +---*I1AOR
      I              I              I              +---BITH1 ----*I1ATST
      I              I              I              +---*I1ASET
      I              I              I              +---PRMCS ----*I1ATST
      I              I              I              +---BITH2 ----*I1ATST
      I              I              I              +---PRMCS ----*I1ATST
      I              I              +---PRMCS ----*I1ATST
      I              +---*I1AAND
      I              +---MCSG3 ----*I1ATST
      I              +---BITHND----*I1ASET
      I              I              +---*I1AOR
      I              I              +---BITH1 ----*I1ATST
      I              I              +---*I1ASET
      I              I              +---PRMCS ----*I1ATST
      I              I              +---BITH2 ----*I1ATST
      I              I              +---PRMCS ----*I1ATST
      I              I              +---MATCH2----*I1AAND

```

```

MAIN  -----MCSGEN-----MCSGET-----MATCH2---+*I1ATST
      I          I          I          +-*I1ASET
      I          I          +---MCSG2
      I          +---MCSSUB-----MCSREF
      I          I          +-*I1ATST
      I          I          +---BITHND-----*I1ASET
      I          I          I          +-*I1AOR
      I          I          I          +---BITH1  -----*I1ATST
      I          I          I          I          +-*I1ASET
      I          I          I          I          +---PRMCS  -----*I1ATST
      I          I          I          +---BITH2  -----*I1ATST
      I          I          I          +---PRMCS  -----*I1ATST
      I          I          +---MATCH2-----*I1AAND
      I          I          I          +-*I1ATST
      I          I          I          +-*I1ASET
      I          I          +---MCSG2
      I          +---MCSAPD
      I          +---MCSOT  -----*I1ATST
      I          I          +---QSORT  -----HEAP  -----HEAPF1
      I          I          +---CUTSET-----*I1ATST
      I          +---RFWRT
      I          +---XCLOCK
      +---MCSOT  -----*I1ATST
      +---QSORT  -----HEAP  -----HEAPF1
      +---CUTSET-----*I1ATST

```

(2) 主要サブルーチンの機能

サブルーチン名	サブルーチンの機能
CUTGT	解析対象フォールトツリーを独立サブツリーと従属サブツリーに分割
DATAIN	入力データの読み込み
INDCHK	独立サブツリーの判定
INISSET	初期設定
INPTRE	フォールトツリーの読み込み
MATCH2	吸収則の判定
MCSGEN	ミニマルカットセットを導出する処理のサブメインルーチン
MCSGET	各サブツリー（中間事象）に対するミニマルカットセットの導出
MCSG1	頂上事象に対するカットセット（中間事象を含む）の導出
MCSNUM	中間事象に対するカットセットの最大個数の計算
MCSOT	計算結果（ミニマルカットセット）の出力
MCS SUB	頂上事象に対するミニマルカットセットの導出 （各サブツリーのミニマルカットセットの合成）
MKMCSF	MCSゲートの入力データ処理
RESET	リスタートファイルからの変数の読み込み（リスタート計算時のみ）
SAMEGT	同一構造のゲートの識別処理
TENKAI	各サブツリーの展開、同一タイプのゲートが連続するレベルで存在する 場合のゲートの処理
TOPDNX	NAND, NOR等の否定型ゲートの識別
TOPDWN	各ゲートの順序付け（頂上事象から順に）
TRANS	NAND, NOR等の否定型ゲートの変換（AND, ORに変換）
TRCOMB	コンビネーションゲートの変換（AND, ORに変換）
TXREF	クロスリファレンスの作成
XCLOCK	CPUタイムの管理及び取り出し

(参考) パラメータ変数の内容

変数名	変数の意味	変数値
YSTCK	作業領域の大きさ	2500
YINCMX	入力コンポーネント最大数	500
YINGMX	入力ゲート最大数	500
YMCSMX	1サブツリーに対する最大カットセット数	20000
YMCSLIN	ミニマルカットセット1個に対するビット表現での文字数 = (YINGMX + YINCMX) / 8	125
YIGCMX	1ゲート当りの入力ゲート/コンポーネント最大数	8
YICGMX	1ゲート/1コンポーネントに対する親ゲートの最大数	10
YIPTMX	ゲート/コンポーネントテーブルの最大数	2000
YLVLMX	独立サブツリー判定の際の最大深さレベル	10
YLIMCS	カットセット吸収則の適用判定割合 (%) (生成されたカットセットの数がYMCSMXの90%になると 吸収則が適用されコア上のMCSテーブルが圧縮される)	90

(3) ジョブ制御文

・ソースプログラムのコンパイル

```
// EXEC FORT77, SO=' Jxxxx. CUTTD', Q=' .FORT77'          --- ソースプログラム
//SYSINC DD DSN=Jxxxx. CUTTDINC.FORT77, DISP=SHR        --- インクルードファイル
// EXEC ASM, SO=' Jxxxx. CUTTD', Q=' .ASM(I1AAND)', DISP=MOD
// EXEC ASM, SO=' Jxxxx. CUTTD', Q=' .ASM(I1AOR)', DISP=MOD
// EXEC ASM, SO=' Jxxxx. CUTTD', Q=' .ASM(I1ASET)', DISP=MOD
// EXEC ASM, SO=' Jxxxx. CUTTD', Q=' .ASM(I1ATST)', DISP=MOD
// EXEC LKEDCT77, LM=' Jxxxx. CUTTD. LOAD', Q=' .LOAD', UNIT=TSSWK
//SYSIN DD *
      ENTRY MAIN
      NAME TEMPNAME
```

アセンブラ
ルーチン

・ロードモジュールからの計算実行

```
// EXEC LMGO, LM=' Jxxxx. CUTTD', PNM=TEMPNAME, A=' COND=(4, LT)'
//FT05F001 DD DSN=Jxxxx. TEST. DATA, DISP=SHR          --- 計算制御データファイル
//FT06F001 DD SYSOUT=I,
      DCB=(LRECL=137, BLKSIZE=145, RECFM=VBA)          --- 計算結果の出力ファイル
//FT13F001 DD DSN=Jxxxx. FT. NF3, DISP=SHR             --- ゲート情報ファイル
//FT14F001 DD DSN=Jxxxx. FT. NF4, DISP=SHR             --- コンポーネント情報ファイル
//FT16F001 DD DSN=Jxxxx. TEST. MCSOUT, UNIT=TSSWK,     --- MCS情報出力ファイル
      SPACE=(TRK, (30, 10), RLSE), DISP=(NEW, CATLG, DELETE),
      DCB=(LRECL=110, BLKSIZE=1100, RECFM=FB, DSORG=PS)
//FT20F001 DD DSN=Jxxxx. TEST. MCS, UNIT=TSSWK,        --- MCS情報ファイル
      SPACE=(TRK, (50, 50), RLSE), DISP=(NEW, CATLG, DELETE), (リスタート用)
      DCB=(LRECL=22000, BLKSIZE=22008, RECFM=VBS, DSORG=PS)
//FT27F001 DD DSN=Jxxxx. TEST. RESTART, UNIT=TSSWK,    --- 制御情報ファイル
      SPACE=(TRK, (50, 50), RLSE), DISP=(NEW, CATLG, DELETE), (リスタート用)
      DCB=(LRECL=22000, BLKSIZE=22008, RECFM=VBS, DSORG=PS)
//FT07F001 DD SYSOUT=I, DCB=(RECFM=FA, BLKSIZE=137)    --- 入力データの出力
```