網羅基準に基づくシーケンス制御プログラムの効率的なテスト技術

Effective Testing Technology for Sequence Control Programs Based on Coverage Criteria

丸地 康平　　進 博正　　吉澤 晋
■MARUCHI Kohei　■SHIN Hiromasa　■YOSHIZAWA Susumu

近年，われわれは様々な社会インフラシステムに支えられ，日々の生活を過ごしている。安全で安心な生活を送るには，これらのシステムが正しく動作することが不可欠であり，高い信頼性が求められる。社会インフラシステムの自動制御には，あらかじめ定められた順序に従って制御の各段階を逐次進めていく，シーケンス制御プログラムがよく使われる。システムの信頼性を高めるには，これらのプログラムに対し十分なテストを行うことが重要となる。

そこで東芝は，発電プラントの制御に使われるプログラミング言語向けに，テストを網羅的に実施するための基準と，この基準を満たすテストを自動生成する技術を開発した。発電プラントで使われているプログラムを用いた評価実験により，この基準を満たしたテストがプログラムの信頼性を高めることを確認した。

A variety of social infrastructure systems have recently become ubiquitous in people's daily lives. High reliability of these systems is necessary to assure safety and security. Sequence control programs that can handle each step of control in a predetermined order are often used for the automatic control of these social infrastructure systems. Sufficient testing of such programs is therefore essential to achieve higher system reliability.

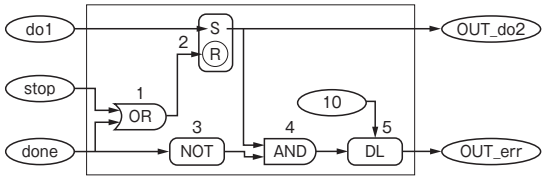
Toshiba has developed a new test coverage criterion for sequence control programs in order to confirm the correctness of sequence control programs used in power plants. As part of this study, we have also developed an automatic test generation technology that generates tests to satisfy coverage criteria. We have conducted evaluation experiments using actual power plant programs and confirmed the effectiveness of our newly developed criterion.

1　まえがき

社会インフラシステムの信頼性を確保することは，われわれが安心して安全に生活するために不可欠である。システムの信頼性を確保するためには，システムが正しく動作することを確認するテストを十分に行うことが重要であり，システムの構成要素であるソフトウェアに対しても，網羅的なテストを実施する必要がある。

C言語に代表される汎用的なプログラミング言語においては，テストを行ううえで満たすべき網羅基準が広く知られ[1]，ソフトウェアの網羅的なテストの実施に活用されている。例えば，全てのプログラムの行を実行する命令網羅や，全ての条件分岐を実行する分岐網羅がある。一方，発電プラントの制御プログラムなど，専用のプログラミング言語を用いて開発されるシステムもある。シーケンス制御プログラムは，論理回路形式で記述されたソフトウェアであり，回路とソフトウェアの両方の性質を持つ。そのため，回路用でもソフトウェア用でもない専用の網羅基準が必要となる。

そこで東芝は，シーケンス制御向けのプログラミング言語に適した専用の網羅基準 MTC (Modified Toggle Coverage) と，この基準を満たすテストを自動生成する技術を開発した。MTCは，論理回路におけるトグル網羅[2]と，ソフトウェアの汎用言語において航空業界で実績のあるMC/DC (Modified Condition/Decision Coverage) の両基準の特徴を兼ね備えた網羅基準である。



do1，stop，done：入力線　　OUT_do2，OUT_err：出力線
10：パラメータ　　S，R：フリップフロップのセット，リセット
DL：遅延　　AND：論理積　　NOT：否定

図1．シーケンス制御言語の例 ― 構成要素となるFBの入出力を信号線で結ぶことでプログラムを作成する。例えば，stop信号とdone信号はORゲートの入力となる。

Example of sequence control program language

ここでは，網羅基準 MTCと，この基準を満たすテストを自動生成する技術について述べる。また，これらの技術が有効であることを確認するため，ミューテーションテスト技法を用い，典型的な不具合の検出率を測定した結果についても述べる。

2　シーケンス制御向けプログラミング言語

ここで対象とするシーケンス制御向けプログラミング言語は，**図1**のように記述される言語である。このような言語は，ファン

クション ブロックダイアグラム（FBD）[3]と呼ばれる。FBDでは，構成要素となるファンクション ブロック（FB）の入出力を信号線で結び，組み合せることでプログラムを構築する。例えば図1のケースでは，論理和（OR）ゲートがFBであり，stop信号とdone信号を入力とし，フリップフロップのリセットに出力している。各FBが内部でどのような演算を行うかは，別途ソフトウェアなどで定義される。

3　新網羅基準 MTC

3.1　従来の網羅基準

網羅基準とは，行われたテストの十分性を確保するために設定する基準である。網羅基準では，テスト中に確認すべき項目を決め，これらの項目を全て満たすかどうかでテストが十分に行われたかを判断する。適切な網羅基準は，テスト対象のプログラム言語に依存し，ソフトウェアや論理回路向けに様々な基準が活用されている。

網羅基準を厳しく設定するほど，テスト対象を十分にテストし信頼性を高めることができるが，基準を厳しく設定しすぎると，実時間で完了できないほどテスト規模が爆発的に増大する。そのため，実用的な規模に収まる範囲で，より厳しい定義の基準ほど良い基準であると言える。

ソフトウェアの場合，全てのプログラムの行を実行する命令網羅（C0），全ての条件分岐を実行する分岐網羅（C1），及び条件分岐の全組合せを実行する複合条件網羅（C2）が知られている[1]。しかし，C0とC1には基準としての弱さがあり，また，C2にはテスト規模が膨大になるという課題がある。これらの課題を解決する基準として，条件分岐の重要な組合せだけを見るMC/DCが知られる。MC/DCは，航空産業におけるソフトウェアの開発ガイドライン DO-178Bに規定される実績のある基準である[1]。MC/DCを満たすには，“プログラムの判定の全条件が判定の出力に独立に影響することを示す”ことが必要になる。

例えば，if（（x==3）or（y>2））{then z^{++};}のプログラムの場合，MC/DCを満たすテストパターンは**表1**のようになる。テストケースとは，テスト入力と期待値の組であり，テストケースの集合体がテストパターンである。

表1．MC/DCを満たすテストパターン
Test pattern covering modified condition and decision coverage (MC/DC)

テストケース			判定（Decision）	条件（Condition）	
ID	x	y	(x==3) or (y>2)	x==3	y>2
t1	−1	2	0	0	0
t2	−1	3	1	0	1
t3	3	2	1	1	0

ID：識別番号

“条件が独立に影響すること”は，他の条件の値を固定し，注目する条件値だけを変え，判定値が変化することを示して確認する。表1の場合，テストケースt1とt3及びt1とt2がそれぞれ，条件（x==3）と条件（y>2）が判定に独立に影響することを確認するテストケースのペアとなる。t1とt3を見ると，（x==3）の値は0から1へ反転し，それ以外の条件である（y>2）は0で等しく，判定は0から1に反転していることが確認できる。同様にt1とt2を見ると，（y>2）の値は0から1へ反転し，それ以外の条件である（x==3）は0で等しく，判定は0から1へ反転していることが確認できる。

また，論理回路向けのカバレッジ基準として，トグル網羅がある。トグル網羅では，論理回路上の信号線に注目し，テスト中にどれだけの信号線で“0から1”と“1から0”への両方の変化が行われたかを見る。FBDは回路図に類似するので，FBD向けの基準としてトグル網羅の適用が考えられる。しかし，トグル網羅の基準の厳しさはソフトウェアのC0やC1相当であるため，MC/DC相当の厳しさを持つ新しい網羅基準が望まれる（**図2**）。

3.2　新網羅基準 MTC

当社はトグル網羅とMC/DCの性質を併せ持つ網羅基準MTCを提案する。条件を入力，判定を出力とみなすことで，MC/DCの“重要な組合せ”だけを確かめる性質をFBDに適用できる。しかし，それでは全ての信号線の動作を確認できるとは限らず，必ずしもトグル網羅より厳しい基準とはならない。そこで，トグル網羅の性質を満たすことを条件に加えることで，基準の厳しさを確保する。具体的には，MTCは，“全ての入力線が出力線に独立に影響することを示す”MC/DCと“全ての信号線で0から1と1から0の両方の値の変化が行われたかを見る”トグル網羅の両方を満たす基準である（**図3**）。

4　網羅基準を満たすテストの自動生成技術

プログラムの規模が大きくなり，構成が複雑になるほど，カバ

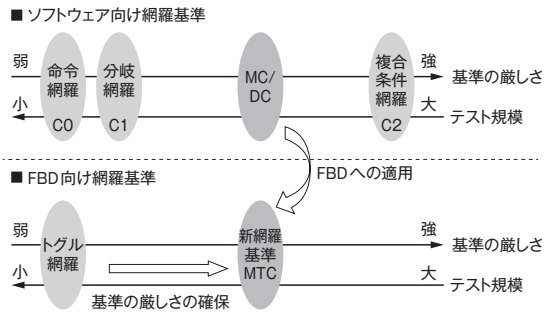


図2．網羅基準の位置関係 ― 新網羅基準のMTCは，FBDにおいて，テスト数を抑えながら基準の厳しさを確保した網羅基準である。
Positional relationships among coverage criteria

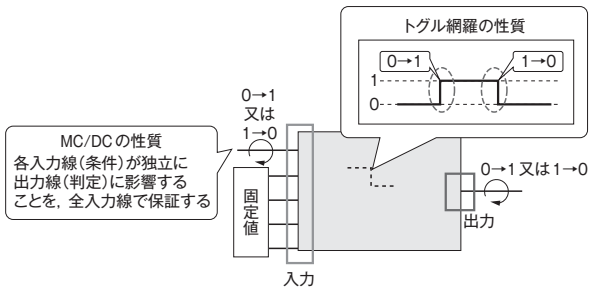


図3．MTCの概念 — MC/DCとトグル網羅の両性質を満たす網羅基準である。

Concept of modified toggle coverage (MTC)

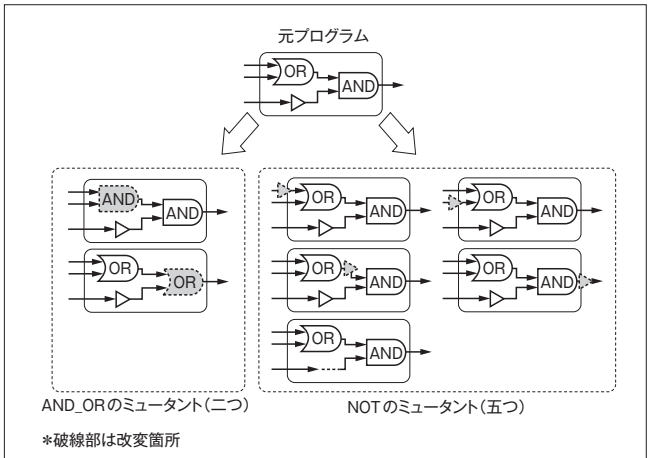


図5．ミュータント生成の例 — プログラムの1か所を改変し，機械的に不具合を混入させたプログラムであるミュータントを生成する。

Examples of mutant generation

レッジ基準を満たすテストパターンを人手で作成することが困難となる。そこで当社は，カバレッジ基準を向上させる制約方程式を算出し，その解を得ることで，網羅基準を満たすテストパターンを自動で生成する技術を開発した。

この技術では，テスト履歴やプログラムから現在の網羅基準の達成度合い（網羅率）を計測し，網羅率を向上させるためにテスト入力が満たすべき方程式を算出する。この方程式を，制約ソルバ[4]を用いて解くことで，網羅率を向上させるテストケースを作成できる。以上の操作を網羅率が100％になるまで繰り返すことで，網羅基準を満たすテストパターンが得られる（**図4**）。

5　新網羅基準の評価

5.1　評価方法

網羅基準の評価は，実際に発電プラントで使われるプログラムを用いて，各網羅基準を満たすテストパターンがどれだけ不具合を見つけることができるかを計測して行った。

評価には，機械的に不具合を混入させるミューテーションテスト技法[5]を用いた。この技法では，元のプログラムに機械的に不具合を混入させたミュータントを生成し（**図5**），これらのミュータントをテストパターンがどれだけ検出できるかでテストパターンを評価する。生成した全てのミュータントのうち，テストパターンが検出したミュータントの割合をミューテーションスコアといい，スコアが高いテストパターンほど機械的に混入された不具合をよく見つけることができるため，優れていると評価できる。

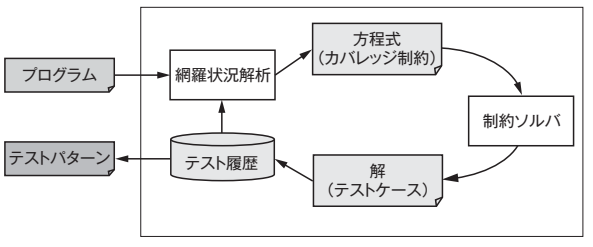


図4．テスト生成の流れ — 制約ソルバを用いて，網羅率を上げるテストケースの生成を繰り返すことで，網羅基準を満たすテストパターンができる。

Block diagram of test generation function

混入させる不具合として，“ANDゲートORゲートの付け間違い”（AND_OR）と“NOTゲートの付け忘れ（誤って付けるのも含める）”（NOT）を用いた。これらは，不具合報告書の調査や設計担当者へのヒアリングを通して得られた典型的な不具合事例である。

評価実験で用いたプログラムは，全て実際の発電プラントの制御プログラムの一部である（**表2**）。命令数は，プログラムに含まれるFBの総数である。AND_ORとNOTは，それぞれに対応した典型的な不具合に相当するミュータントの総数である。網羅基準を満たすテストパターンは，前述した自動生成技術を用いて作成した。網羅基準としては，MTC，MC/DC，及びトグル網羅を用いた。

5.2　評価結果

ミューテーションスコアの計測結果を**図6**に示す。MTC，MC/DC，及びトグル網羅のそれぞれのスコアを平均すると0.985，0.967，及び0.872となり，スコアの高さはこの順番になる傾向を確認した。トグル網羅とMC/DCのスコアの差に比べ，MC/DCとMTCのスコアの差が小さいことから，MC/

表2．評価に用いたプログラム

Example of programs for evaluation experiments

プログラム					ミュータント	
ID	命令数	入力数	状態数	出力数	AND_OR	NOT
P1	165	36	19	26	51	162
P2	158	37	14	25	48	155
P3	115	24	10	23	33	104
P4	115	14	7	11	20	58
P5	113	29	10	18	34	103

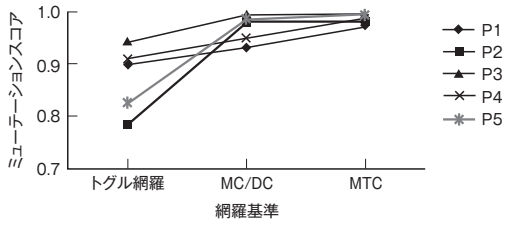


図6. 評価結果：網羅基準の比較 ― MTCを満たすテストパターンが，どのプログラムに対してももっとも高いスコアとなり，典型的な不具合の検出能力が高いことを確認できた。

Results of comparative evaluation of coverage criteria

DCが特に有効に働いていることがわかる。これは，出力への影響を見るMC/DCの性質が効いていると考えられ，その理由は，不具合を検出するには，ふるまいの違いが出力に表れる必要があるためである。トグル網羅は個々の信号線の値の変化しか見ないため，その変化が出力に表れるとは限らず，不具合箇所を実行しても，ふるまいの違いが検出できないケースがあると考えられる。

また，プログラムP1やP4において，MC/DCとMTCのスコア結果に0.04程度の差が出る結果となった。これは，トグル網羅の値の変化を見る性質が効いていると考えられる。トグル網羅では，0から1のように信号線の値の変化を見るため，着目している信号値を変化前の値にするためのテスト入力と，変化後の値にするためのテスト入力を連続して実行する必要がある。一方，MC/DCでは必ずしも連続して行う必要がない。プログラムP1やP4では，内部状態の値の変化を通して，出力の値が変わるミュータントが生成されている。連続してテストを実施するトグル網羅のほうがこれらのミュータントを検出しやすいと考えられる。

続いて，プログラムP5に対し，テスト入力を乱数から求めたランダム テストパターンを加えて評価した結果が**図7**である。この結果から，ランダム生成したテストでは，テストケース数を100から800へ8倍に増やしてもミューテーションスコアは0.05

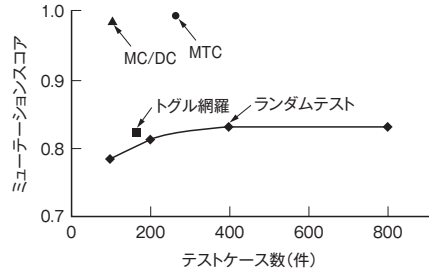


図7. 評価結果：ランダムテストとの比較 ― ランダムテストの数を増やしてもスコアが上がらないことから，網羅基準を用いたテストでは，テスト数を抑えながら典型的な不具合の検出率が高いことを確認できた。

Results of comparative evaluation of random testing

程度しか向上せず，やみくもにテスト数を増やしても，効率的に不具合は見つけられないことを確認できた。これにより，MTCを満たすテストパターンがより多くのミュータントを検出したのは，単にテスト数が増えたからではなく，両方の基準の性質を併せ持つことによるためと考えられる。すなわち，新網羅基準MTCの有効性を示した結果であると言える。

6 あとがき

シーケンス制御プログラミング言語に適した新しい網羅基準 MTCと，この基準を満たすテストパターンを自動生成する技術について述べた。また，発電プラントで実際に使われるプログラムをモチーフとした評価実験を行い，MTCを満たすテストパターンがプログラム品質確保に有効であることを示した。

今後は，更に効果の高いカバレッジ基準や，テスト数を抑えたテスト生成技術の開発に取り組む。

文 献

(1) Chilenski, J. J. ; Miller, S. P. Applicability of modified condition/decision coverage to software testing. Software Engineering Journal. **9**, 5, 1994, p.193 - 200.

(2) Kantrowitz, M. ; Noack, L.M. "I'm Done Simulating; Now What? Verification Coverge Analysis and Correctness Checking of the DECchip 21164 ALPHA microprocessor". Proceedings of the 33rd Design Automation Conference. Las Vegas, NV, USA, 1996-06, ACM. 1996, p.325 - 330.

(3) IEC 61131-3 ed. 3.0: 2013. Programmable controllers – Part 3: Programming languages. International Electrotechnical Commission.

(4) Rossi, F. et al. Handbook of Constraint Programming (Foundations of Artificial Intelligence). Elsevier Science, 2006, 978p.

(5) DeMillo, R. et al. Hints on Test Data Selection: Help for the Practicing Programmer. Computer. **11**, 4, 1978, p.34 - 41.

丸地 康平 MARUCHI Kohei
研究開発センター システム技術ラボラトリー研究主務。
ソフトウェアの検証及びシステムの高信頼化技術の研究・開発に従事。
System Engineering Lab.

進 博正 SHIN Hiromasa
研究開発センター システム技術ラボラトリー主任研究員。
システム及びソフトウェア工学分野の研究・開発に従事。情報処理学会会員。
System Engineering Lab.

吉澤 晋 YOSHIZAWA Susumu
電力システム社 府中電力システム工場 電力プラットフォーム開発部主務。発電所向け監視制御コントローラTOSMAP™の設計・開発に従事。
Fuchu Operations - Power Systems