ソフトウェア工学
11章　形式手法

発表者：M1 倉地亮介

目次

- 形式手法の概要
  1) 形式手法とは　2) 形式手法のねらい　3) 論理体系の例
- 形式仕様と検証
  1) 形式仕様　2) 形式検証
- モデル検査
  1) モデル検査とは　2) 仕様記述　3) 時相論理
  4) モデル検査による検証　5) 状態爆発
- 形式手法の活用
  1) モデルの構築と利用　2) 開発の中での位置づけ
- まとめ

形式手法の概要

形式手法とは？

数理論理学に基づいて対象や性質の記述を行うことで、
システムの開発や検証を体系的に行う手法

- 数理論理学: 論理を記号化，形式化し，数学的手法に基づいて研究する論理学

形式手法適用事例

項番	分野	適用件数
1	航空	4
2	宇宙	2
3	原子力	2
4	鉄道	11
5	船舶	1
6	公共	1
7	金融	1
8	医療	1
9	自動車	2
10	社内システム	2
11	セキュリティ	2
12	OS	2
13	ネットワーク	3
14	ICカード	3
15	携帯電話	2
16	ハードウェア	2
17	組込み機器	2
18	電力	14
		57

【出典】IPA/SEC高信頼性システム開発技術の動向
第3章「形式手法の適用事例」

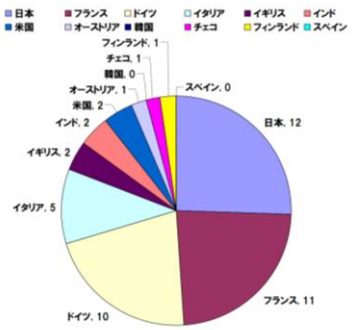
項番	分野	適用件数
1	鉄道	36
2	航空宇宙	26
3	医療機器	2
4	医療	2
5	産業	5
6	金融	2
7	エンタプライズ	2
8	防衛	3
9	公共	2
10	通信	1
11	セキュリティ	5
12	半導体	1
13	原子力	3
14	その他	1
15	?	9
		100

【出典】IPA/SEC「形式手法適用調査報告書」
（2010年7月29日公開）

業界別

項番	国	適用件数
1	日本	12
2	フランス	11
3	ドイツ	10
4	イタリア	5
5	イギリス	2
6	インド	2
7	米国	2
8	オーストリア	1
9	韓国	0
10	チェコ	1
11	フィンランド	1
12	スペイン	0

【出典】高信頼性システム開発技術の動向
第3章「形式手法の適用事例」

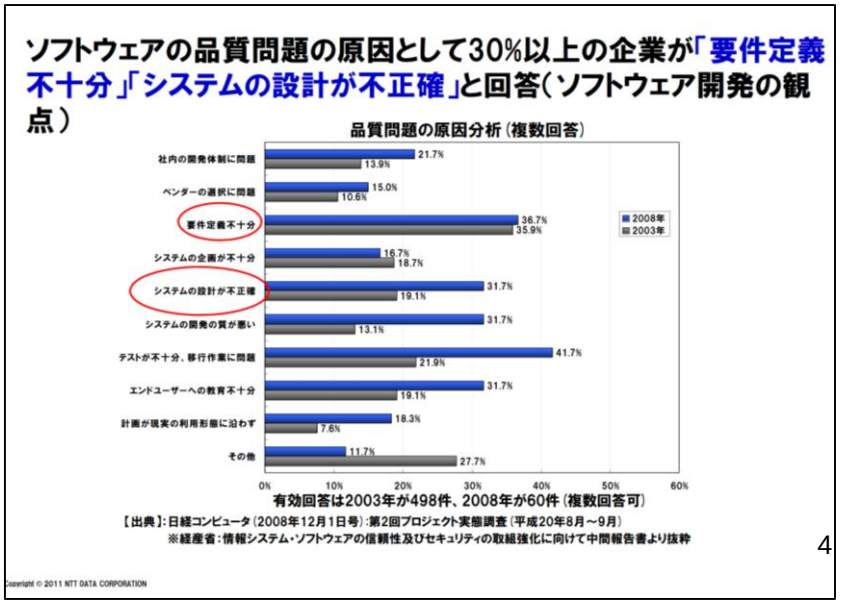


国別

形式手法のねらい

- 定義される対象を厳密に記述し，何らかの論理体系に基づいて，それの正しさについて議論を行うこと
  - e.g.ソフトウェアの要求仕様，設計，プログラム

- 形式手法を用いることのメリット(1)
  - 対象を厳密に記述できる
  - 正しさを厳密に議論できる
  - 不具合を発見できる
  - 不具合がないことを確認できる



論理体系の例(1)

- 命題論理
  - 真か偽かどちらかであるような言明(命題)に対する正しい推論形式

¬	否定（でない）
∧	連言（かつ）
∨	選言（または）
→	含意（ならば）
⇔	同値

図 11.1　論理記号

表 11.1　真理表

A	B	A∧B	A∨B	A→B	A⇔B
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

- 真理表を構成的に適用することで，任意の論理式の真偽を決定できる

  e.g.　(A∨B)∧C　　[A : T , B : F , C : T]

何らかの対象の正しさをではなく，論理式の正しさを議論するもの

論理体系の例(2)

- 述語論理
  - 個体の性質や関係を表す表現(述語)に対する正しい推論形式

e.g.「xは0より大きい」　　x= 3 ：真
x=-1 ：偽

  - 対象の選び方で真偽が変わることがある

∀x P(x) 全称記号	すべてのxについてP(x)が真
∃x P(x) 存在記号	P(x)を真にするxが存在

図 11.2　量化記号

e.g. $\forall x(x \geq 0)$

論理式だけでは真偽が決まらず，論理式と対象との対応づけが決まって初めて真偽が議論できる

形式仕様

- ソフトウェア工学における仕様
  - システム定義や検証を目的としたシステムに関する明確な記述
    - e.g. 要求仕様，設計仕様，製品の仕様，テスト仕様　etc …
- 形式手法における仕様
  - 設計などの対象を，論理体系に基づく記述方法で記述した仕様を形式仕様と呼ぶ
    - 操作的仕様：対象の望まれる振る舞いを記述する方法

      e.g. ステートマシン図
    - 記述的仕様：対象の望まれる性質を宣言的に記述する方法

      e.g. 実体関連図

形式検証

- 仕様が記述されると，論理体系の枠組みの中で，その正しさについて議論することができる

■ 定理証明：数学の証明と同様に，推論規則を適用して性質が成立することを証明する技術

- 自動定理証明のためのツール
  - あらかじめ推論規則や論理式の変形方法などを備えている

■ モデル検査：仕様記述から導出される状態空間を全数探索することで，性質の確認を行う

一般的に，定理証明の方がその適用に専門スキルが必要なため、
モデル検査の方が利用の敷居が低く，広く使われている

モデル検査

モデル検査とは

仕様記述から導出される状態空間を全数探索することで，性質の確認を行う

- 検証対象：有限状態モデルを用いて記述され，性質記述には時相論理が用いられる

モデル検査の概要

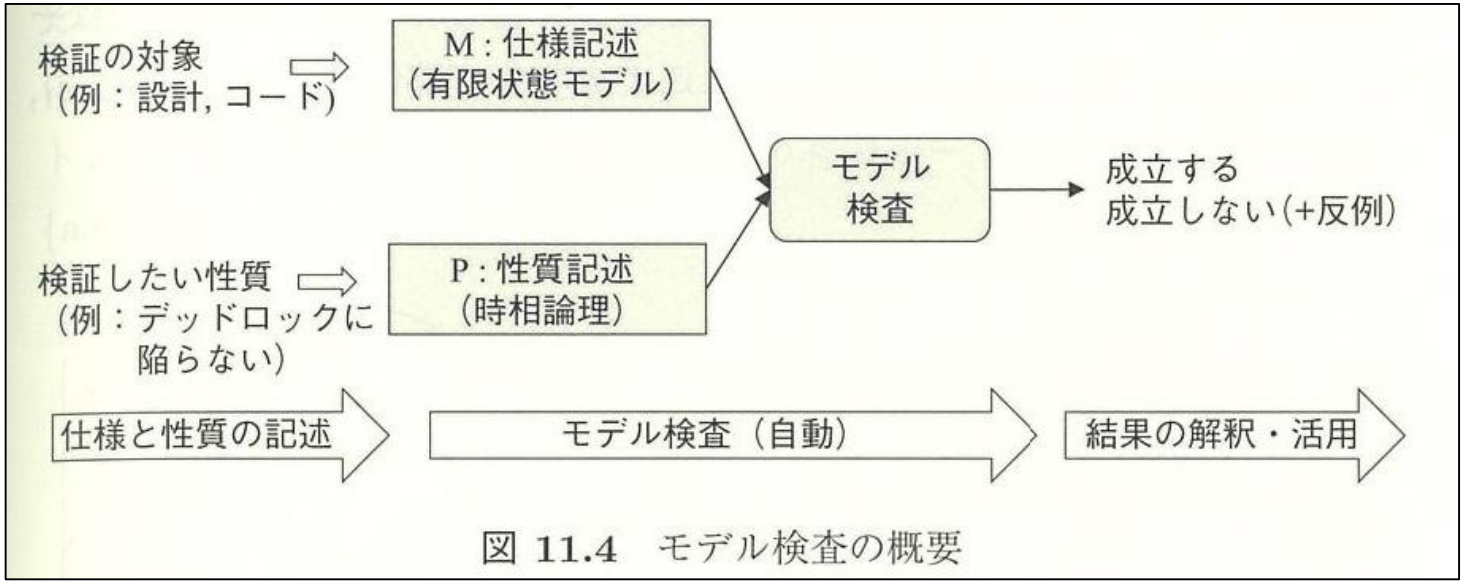


図 11.4　モデル検査の概要

モデル検査-仕様記述(1)

クリプキ構造

- ラベル付けされた遷移状態中の式で評価される非決定性有限オートマトン
- 4つ組で定義される

S: 状態の有限集合

S_0: 初期状態の集合 ($S_0 \subseteq S$)

R: 遷移の集合 ($R \subseteq S \times S$)

L: ラベリング関数 ($S \rightarrow 2^{AP}$)

- 原子命題の集合をAP
- ラベリング関数は各状態で真となる原子命令を与える関数

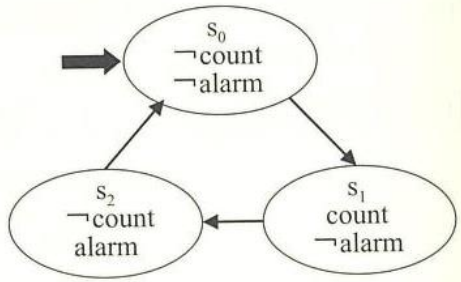
クリプキ構造の例

$S = \{s_0, s_1, s_2\}$

$S_0 = \{s_0\}$

$R = \{ (s_0, s_1), (s_1, s_2), (s_2, s_0) \}$

$L = ((s_0, (\neg count, \neg alarm)),$
$(s_1, (count, \neg alarm)),$
$(s_0, (\neg count, alarm)))$



モデル検査-仕様記述(2)

オートマトン

- イベントに対してシステムがどのように状態を変化させるかを表すためのモデル
- 5つ組で定義される

S: 状態の有限集合
S_0: 初期状態の集合($S_0 \subseteq S$)
F: 受理状態の集合($F \subseteq S$)
Σ ラベル（イベント）の有限集合
δ 遷移の集合($\delta \subseteq S \times \Sigma \times S$)

- 正規表現:有限オートマトンが受理する言語

$S = \{s_0, s_1, s_2\}$
$S_0 = \{s_0\}$
$F = \{s_2\}$
$\Sigma = \{a, b\}$
$\delta = \{(s_0, a, s_1), (s_1, b, s_0), (s_1, a, s_2)\}$

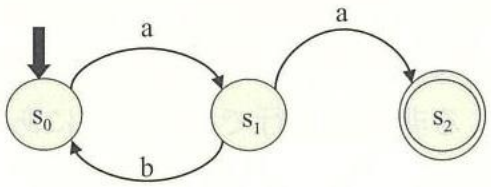


図 11.8 有限オートマトンの例

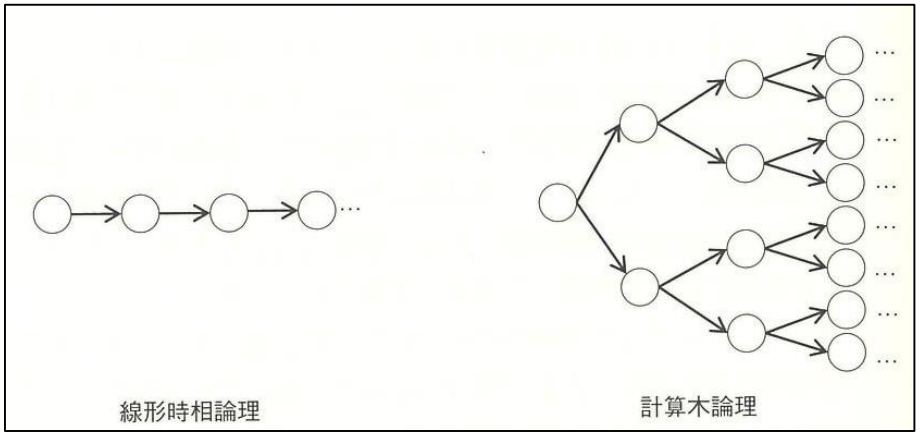
時相論理

- 命題の真偽が時間に依存する論理　※離散的な時間のみを扱う

  ー時間との関連で問題を理解し表現するための規則と表記法の体系

  e.g.「私は腹ペコだ」

- 時間の捉え方
  - 線形時相論理
  - 計算機論理



$X\psi$	(neXt)
$F\psi$	(Future)
$G\psi$	(Globally)
$\psi U\phi$	(Until)
$\psi R\phi$	(Release)

時間に関する論理記号

モデル検査による検証(1)

◆ 例題

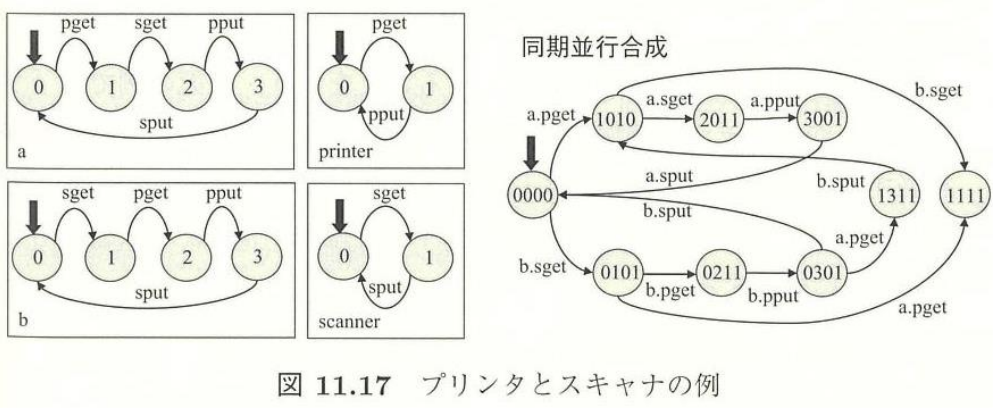


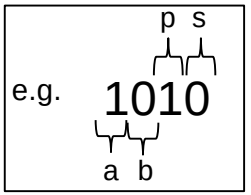
図 11.17　プリンタとスキャナの例

概要

- プロセスa，プロセスb，プリンタ，スキャナ
- 4つは並列に動作
- 2つのプロセスはプリンタやスキャナを利用する際には，利用権を確保
- 利用後には，利用権を開放する

■ 仕様記述

- 各プロセスとプリンタ，スキャナの振る舞いを示す



■ 性質記述

1. 安全性：ある状態に決して陥らないという性質
2. 活性　：ある条件が成立すれば必ずある状態に至るという性質
3. 到達可能性：初期状態から特定の状態に到達しうること

モデル検査による検証(2)

◆ 例題

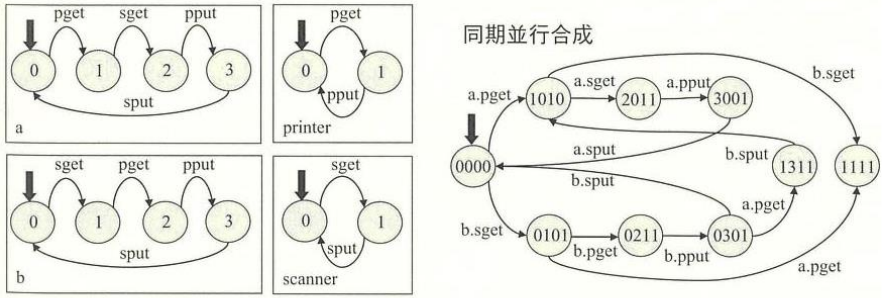


図 11.17　プリンタとスキャナの例

■ 性質記述

✓ 安全性
✓ 到達可能性
× 活性 → 反例 [0000]→[1010]→[1111]

デッドロックが発生する

◆ 例題(改良版)

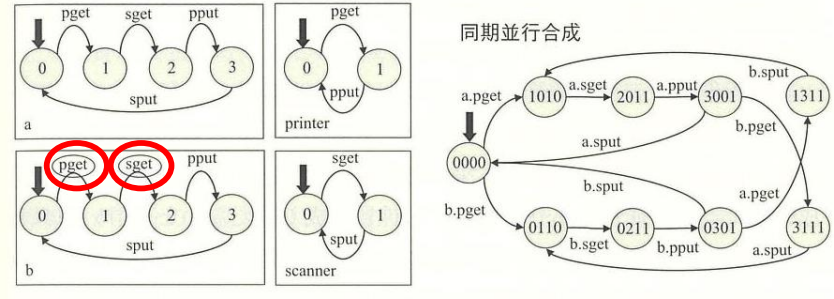


図 11.18　プリンタとスキャナの例（改良版）

改良点
● プロセスbが利用権をとる順番を入れ替える

■ 特定の並行動作単位のみが実行され続ける状況が発生する

e.g. [0000]→[1010]→[2011]→[3001]→[0000]→…

公平性:実行可能なプロセスは無限回実行されるという性質

状態爆発

- 検査する状態数が大きくなり，現実的な時間やメモリ量で検査が終了しない状態

- 状態爆発への対応方法
  - 抽象化：検証対象の状態数を減らすように仕様記述を抽象化する方法
  - 部分的探索：検証の範囲を限定して，その範囲の中だけの状態を探索する方法

    e.g. 有界モデル検査

形式手法の活用

- モデルの構築と利用
  - 形式手法はソフトウェアそのものを直接確認する技術ではなく，モデル化して，モデルに対して検証を行う

    モデル上で確認できた性質が，現実世界のおけるソフトウェアでも成立するかどうかということが重要

    - モデル化，性質記述する際に，それらが妥当性か？
    - モデル検証の結果が分かった時それをどう解釈するか？

- 開発の中での位置づけ

  ーどのタイミングで，何を対象に形式手法を適用すればいいのか？

形式手法は適用コストが高いため，開発全体を見てその適用の目的や効果を考える視点がより大切

まとめ

- 形式手法の概要

  数理論理学に基づいて対象や性質の記述を行うことで，
  システムの開発や検証を体系的に行う手法

- 形式仕様と検証

  - 形式仕様の記述方法
  - 記述が正しいことを証明

- モデル検査

  モデル検査の概要，検証の特性，また利用する際の重要な課題である状態爆発

- 形式手法の活用

  - モデルの構築と利用
  - 開発の中での位置づけ