プログラムの理論

計算についての我々の理解を数学的に厳格にすること,
なかんずくコンピュータ・プログラムについての検証のアート
(かの名高いデバッグの技術)を1つのサイエンスにすることを目指す理論
Z. Manna (1974), Mathematical Theory of Computation, McGraw-Hill.
“I consider it to be the theory which attempts to formalize our understanding of computation, and in particular to make the art of verifying computer programs (the famous debugging technique) into a science.”

呼び方 (いずれも1960年代初頭より)

日本	Theory of Program (Program Theory)	Igarashi
アメリカ	Mathematical Theory of Computation	McCarthy
ソ連(ロシア)	Theoretical Programming	Ershov

(このprogrammingはsoftware engineering, software science, computer scienceといった意味.)

whileプログラム

逐次プログラムの基本的なコントロールを網羅した抽象的言語

関数呼び出し，goto，変数の型などはない

部分的正当性（実行前と実行後の変数の値の関係），および停止性に関する議論を行う

　完全正当性＝部分的正当性＋停止性

プログラムの検証

プログラムが正しく作動することを確認する作業

検証方法は数学的・論理的に厳密にしかつ一般性を持たせる

逐次プログラムの検証

Hoare論理

- 対象：単純な構造的プログラム言語
- 検証：部分的正当性, 停止性

BNF(Backus Naur Form)による言語の定義

式(expression)

5, x, 4+(x-3), x+(x*(y-(5+z)))など

BNF(Backus Naur form)での定義

E::=n | x | (-E) | (E+E) | (E-E) |(E*E) |(E/E)|…

但し n: 数値, x: 変数

括弧は適宜省略する

ブール式(Boolean Expression)

B::= true | false | (!B) | (B & B) | (B || B) | (E<E)

但し, ! : 否定, & : 論理積, || : 論理和

等号==は!(E1<E2)&!(E2<E1)で定義できる

(E1!=E2)は!(E1==E2)の略記

文(statement)

C::= x=E｜C; C｜if B {C} else {C}｜while B {C}

x=E

割当文(代入文)(assignment statement)

if B {C_1} else {C_2}

条件文(conditional statement)

while B {C}

while文(while statement)

C_1; C_2

連結文(compound statement)

if B {C_1}

も文に含める

(何の略記と考えれば良いか？)

プログラム(program)

P::= C

例

```
x=a; y=b;
while (y!=0) {x=x mod y; z=x; x=y; y=z};
if (x<0){x=-x}

x=a; y=1; z=0;
while (z!=x){z=z+1; y=y*z}
```

プログラムの意味

正当性を正しく厳密に議論するためには,

対象となるプログラムの「意味」(semantics)を与えなければならない.

プログラムの実行（Execution）

ρ : 状態(state)

プログラム変数xから値 $\rho(x)$ への関数

プログラムの実行で ρ が変化する　$\rho \to \rho'$: 遷移

プログラムの実行(execution)

$\rho_0 \to \rho_1 \to \rho_2 \to \rho_3 \to \ldots \to \rho_i \to \rho_{i+1} \to \ldots$

ρ_0 : 初期状態

プログラムの実行関係（Execution Relation）

関係$Exec_M$(P, ρ, ρ')でプログラムPの実行によるρからρ'への遷移を与える.

M: モデル

$Exec_M$(x=t, ρ, ρ')　⇔ $\rho' = \rho[M[t]_\rho / x]$

すなわち $\rho'(y) =$ 　$\rho(y)$　if yが変数xでないとき

$M[t]_\rho$　if yがxのとき

ここで$M[t]_\rho$はモデルMおよび状態ρのもとでのtの値

$Exec_M$(if B {C_1} else {C_2} , ρ, ρ') ⇔　M, $\rho \models B \Rightarrow Exec_M(C_1, \rho, \rho')$

M, $\rho \not\models B \Rightarrow Exec_M(C_2\ \rho, \rho')$

（M, $\rho \models$B : モデルMと状態ρのもとでBが成立つ）

関係 $\mathrm{Exec}_M(P, \rho, \rho')$でプログラムPの実行による ρから ρ'への遷移を与える

$\mathrm{Exec}_M(C_1; C_2, \rho, \rho')$ ⇔

ある ρ_1が存在して $\mathrm{Exec}_M(C_1, \rho, \rho_1)$かつ$\mathrm{Exec}_M(C_2, \rho_1, \rho')$

$\mathrm{Exec}_M(\text{while } B \ \{C\}, \rho, \rho')$ ⇔

ある$m>0$および $\rho_1,\ldots, \rho_m$が存在して，$\rho=\rho_1$, $\rho'=\rho_m$,

$M, \rho_m \not\models B$，m未満の全てのiで $M, \rho_i \models B$かつ $\mathrm{Exec}_M(C, \rho_i, \rho_{i+1})$

例題

Succを以下のプログラムとする.

```
a=x+1; if (a-1==0){y=1}else {y=a}
```

Mを自然数に関する通常の解釈とし, ρを$\rho(x)=0$を満たすものとする.

このとき, $\mathrm{Exec}_M(\mathrm{Succ}, \rho, \rho')$が真であるような ρ'を示せ.

解

Exec_M(a=x+1; if (a-1==0){y=1}else {y=a}, ρ, ρ'）

⇔ ある ρ"が存在して

Exec_M(a=x+1, ρ, ρ")かつ

Exec_M(if (a-1==0){y=1}else {y=a}, ρ", ρ'）

ここでExec_M(a=x+1, ρ, ρ")を満たす ρ"は

ρ"(z)=	ρ(z)	if zが変数aでないとき
	1	if zがaのとき

である.

M, ρ"⊨a-1=0であるため, 元の式⇔Exec_M(y=1, ρ", ρ')であり, これを満たす ρ'は

ρ'(z)=	ρ(z)	if zが変数aでもyでもないとき
	1	if zがaまたはyのとき

となる. 特に, 問題の前提より ρ'(x)= ρ(x)=0である.

演習問題

以下のプログラムを考える.

while x<10 {x=x+1}

Mを自然数に関する通常の解釈とし，ρをρ(x)=0を満たすものとする.

このとき, Exec_M(while x<10 {x=x+1}，ρ，ρ’)が真であるような ρ’を示せ.

演習問題

以下のプログラムを考える.

while true {x=0}

Mを自然数に関する通常の解釈とする.

このとき, $Exec_M$(while true {x=0}, ρ, ρ’)が真であるような対<ρ, ρ’>を示せ.

Hoare論理概説

1960年代に考案された手続き型プログラムのための形式的証明体系

C. A. R. Hoare (1969), "An axiomatic basis for computer programming", Communications of the ACM 12(10): 576-580,583. http://sunnyday.mit.edu/16.355/Hoare-CACM-69.pdf

現在でもプログラム検証の基礎として有効である

ほとんどの検証体系はこの論理に影響を受けている

形式的検証体系(formal system)なので,

プログラムを(前に述べたようなプログラムの意味を考えるのではなく)

形式的に検証することができる

表明付きプログラム
Hoareの3つ組 (Hoare triples)

《 ϕ 》P《 ψ 》

P:プログラム(program)

ϕ, ψ: 表明 (assertion)

P中のプログラム変数の関係式(論理式)

意味

Pの実行直前に ϕが成立するならば, Pの実行結果は ψを満たす.

ϕ: 前条件(precondition), ψ: 後条件(postcondition)

表明付きプログラム
Hoareの3つ組 (Hoare triples)

Hoareによる原論文では $\phi\{P\}\phi$の形
現代では$\{\phi\}P\{\phi\}$
本講義ではCのような“{” “}”を用いるプログラムを
対象にするためこの表記を使わない

$\langle\!\langle \phi \rangle\!\rangle P \langle\!\langle \phi \rangle\!\rangle$

P:プログラム(program)

ϕ, ϕ: 表明 (assertion)

P中のプログラム変数の関係式(論理式)

意味

Pの実行直前に ϕが成立するならば, Pの実行結果は ϕを満たす.

ϕ: 前条件(precondition), ϕ: 後条件(postcondition)

例

P: 二乗がx未満になる数を計算するプログラム, 但しxは正

$$⟪ x>0 ⟫ P ⟪ y^2<x ⟫$$

例えばPが単にy=0でもok.

y=0; while (y*y<x){ y=y+1 }; y=y-1でもok.

例

P: 二乗がx未満になる最大の数を計算するプログラム, 但しxは正

$\langle\!\langle\, x>0 \,\rangle\!\rangle$ P $\langle\!\langle\, y^2<x \leq (y+1)^2 \,\rangle\!\rangle$

例えばPが

```
y=0; while (y*y<x){ y=y+1}; y=y-1
```

であればok.

例

P: 二乗がx未満になる最大の数を計算するプログラム, 但しxは正

$\langle\!\langle\, x>0 \,\rangle\!\rangle$ P $\langle\!\langle\, y^2<x \leq (y+1)^2 \,\rangle\!\rangle$

例えばPが

```
y=0; while (y*y<x){ y=y+1}; y=y-1
```

であればok.

厳密には, プログラム開始時と終了時の変数の値を区別できるようにした方が良い(後述)

部分的正当性
（partial correctness）

定義

$(\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$ が部分的に正当である partially correct

$\Leftrightarrow$

ϕが成立つどんな状態でPを実行しても,

Pの実行が終了するならば実行後の状態で ψが成立つ.

すなわち, どんな ρ, ρ' に対しても$M,\rho \vDash \phi$かつ$\mathrm{Exec}_M(P, \rho, \rho')$ならば$M,\rho' \vDash \psi$

$\vDash_{par} (\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$と表す.

部分的正当性
（partial correctness）

ここでの ρ, ρ' は
プログラム変数に対する値の割当(附値)で
あることに注意
(P中には自由変数は存在しない.
ϕ, ψに自由変数がある場合は後述)

定義

$⦅\phi⦆P⦅\psi⦆$ が部分的に正当である partially correct

$\Leftrightarrow$

ϕが成立つどんな状態でPを実行しても,

Pの実行が終了するならば実行後の状態で ψが成立つ.

すなわち, どんな ρ, ρ' に対してもM,$\rho \models \phi$かつ$\mathrm{Exec}_M(P, \rho, \rho')$ならばM,$\rho' \models \psi$

$\models_{par} ⦅\phi⦆P⦅\psi⦆$と表す.

完全正当性 （total correctness）

定義

⦅ ϕ ⦆P⦅ ψ ⦆ が完全に正当である totally correct

⇔

ϕが成立つどんな状態でPを実行しても, Pの実行は終了して, 実行後の状態で ψが成立つ.

すなわち, どんな ρ に対してもM, $\rho \vDash \phi$ ならばある ρ' が存在してExec_M(P, ρ, ρ')

かつM, $\rho' \vDash \psi$

$\vDash_{tot}$ ⦅ ϕ ⦆P⦅ ψ ⦆と表す.

例

どんな ϕ, ψに対しても

$$\vDash_{par} (\!|\ \phi\ |\!) \text{while true } \{x=0\} (\!|\ \psi\ |\!)$$

は成立つ. $\vDash_{tot}$にすると成立たない.

(註)プログラムが停止しないというのは上記のようなループが停止しない場合のみを示す.

Succを以下のプログラムとする.

a=x+1; if (a-1==0){y=1}else {y=a}

$(\!|\ \top\ |\!)$ Succ $(\!|\ y=(x+1)\ |\!)$は部分的正当, 完全正当ともにいえる.

演習

解釈Mを, 整数に関する標準的な解釈とする.

以下が成り立つことを示せ.

1. $\models_{par} ⦅ x>0 ⦆ y=0 ⦅ y^2<x ⦆$

2. $\models_{tot} ⦅ x=0 ⦆$ while x<10 {x=x+1} $⦅ x=10 ⦆$

3. どんな ϕ, ψに対しても $\models_{par} ⦅ \phi ⦆$while true {x=0}$⦅ \psi ⦆$

どんな ϕ, ψに対しても以下が成り立たないことを示せ.

4. $\models_{tot} ⦅ \phi ⦆$while true {x=0}$⦅ \psi ⦆$

プログラム変数と論理変数

プログラム変数 Program variables	検証対象のプログラムの変数
論理変数 Logical variables	表明のための変数, プログラム中に出現しない

例

Fac$_2$: y=1; while (x!=0) {y=y*x; x=x-1}

このプログラムの部分的正当性は

$$⦅ x \geq 0 ⦆ \mathrm{Fac}_2 ⦅ y = x! ⦆$$

ではなく

$$⦅ x = x_0 \wedge x_0 \geq 0 ⦆ \mathrm{Fac}_2 ⦅ y = x_0! ⦆$$

x_0: 論理変数(logical variable)

論理変数を含むHoareの3つ組が部分的正当

$$\vDash_{par} (\!|\ \phi\ |\!) P (\!|\ \psi\ |\!)$$

であるとは

論理変数に対するどのような附値 $\rho^{\llcorner}$および,

プログラム変数に対するどのような状態(附値) ρ, ρ' に対しても

$M, \rho^{\llcorner}, \rho \vDash \phi$かつ$Exec_M(P, \rho, \rho')$ならば $M, \rho^{\llcorner}, \rho' \vDash \psi$

完全正当性の場合も同様.

形式的体系 (formal system) [再掲]

公理 axioms

　妥当(valid)な命題

推論規則 inference rules

　一つ以上の妥当な命題から他の妥当な命題への写像

命題が証明可能(provable), 演繹可能(deducible), 定理(theorem)

　公理, 推論規則の適用の繰り返し(証明(proof))から得られる命題

部分的正当性に関するHoare論理の体系で

$(\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$ が証明できたとき, $\vdash_{\mathrm{par}} (\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$と表す.

完全正当性に関するHoare論理の体系で

$(\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$ が証明できたとき, $\vdash_{\mathrm{tot}} (\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$と表す.

$\vdash_{\mathrm{tot}} (\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$ならば $\vdash_{\mathrm{par}} (\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$

$\vDash_{\mathrm{tot}} (\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$ならば $\vDash_{\mathrm{par}} (\!|\ \phi\ |\!)P(\!|\ \psi\ |\!)$

体系が健全(sound):

その体系で証明できる命題は必ず正しい(valid).

Hoare論理では

$\vdash_{par}$ ⦅ ϕ ⦆P⦅ ψ ⦆ならば必ず $\vDash_{par}$ ⦅ ϕ ⦆P⦅ ψ ⦆

$\vdash_{tot}$ ⦅ ϕ ⦆P⦅ ψ ⦆ならば必ず $\vDash_{tot}$ ⦅ ϕ ⦆P⦅ ψ ⦆

体系が完全(complete)

正しい命題は必ず証明できる.

Hoare論理では

$\vDash_{par}$ ⦅ ϕ ⦆P⦅ ψ ⦆ならば必ず $\vdash_{par}$ ⦅ ϕ ⦆P⦅ ψ ⦆

$\vDash_{tot}$ ⦅ ϕ ⦆P⦅ ψ ⦆ならば必ず $\vdash_{tot}$ ⦅ ϕ ⦆P⦅ ψ ⦆

実際はHoare論理は相対完全(relative complete)

用いている数学が完全ならばHoare論理も完全

レポート問題

解釈Mを, 整数に関する標準的な解釈とする.

Fac_1を以下のプログラムとする. (変数は整数値をとるものとする.)

```
y=1; z=0; while (z!=x){z=z+1; y=y*z}
```

このとき

$$\models_{tot} \langle\!\langle\ x=x_0 \wedge x_0 \geq 0\ \rangle\!\rangle\ Fac_1\ \langle\!\langle\ y=x_0!\ \rangle\!\rangle$$

が真であることを示せ. ただしx_0 は論理変数である.

締切　7月23日(水)15:15

提出先　manaba