Owicki-Gries Logic for Concurrent Programs

Motivation

- Concurrency introduce non-determinism
  - Scheduling strategies are intentionally under-specified
  - We cannot predict the exact order of concurrent commands

- State sharing between threads makes modular proofs difficult
  - Disjointness / ownership helps to *split* proof obligations

- Surprising discovery: *locks* introduce *invariants* about ME accessed resources
  - ME — mutual exclusion
  - Somewhat similar to loop invariants

Extending Hoare Logic for Concurrency: 1976

https://en.wikipedia.org/wiki/Susan_Owicki

https://en.wikipedia.org/wiki/David_Gries

Operating Systems
R.S. Gaines Editor

Verifying Properties of Parallel Programs: An Axiomatic Approach

Susan Owicki and David Gries
Cornell University

An axiomatic method for proving a number of properties of parallel programs is presented. Hoare has given a set of axioms for partial correctness, but they are not strong enough in most cases. This paper defines a more powerful deductive system which is in some sense complete for partial correctness. A crucial axiom provides for the use of auxiliary variables, which are added to a parallel program as an aid to proving it correct. The information in a partial correctness proof can be used to prove such properties as mutual exclusion, freedom from deadlock, and program termination. Techniques for verifying these properties are presented and illustrated by application to the dining philosophers problem.

Key Words and Phrases: structured multiprogramming, correctness proofs, program verification, concurrent processes, synchronization, mutual exclusion, deadlock

CR Categories: 4.32, 4.35, 5.21, 5.24

Simple Language with Concurrency

- Programming language derived from Algol 60
- Syntactic notation
  - r – a set of variables
  - S – a statement
  - B – a boolean condition
- Parallel execution statement:

  resource $r_1,\ldots,r_m$:
  cobegin S_1 || … || S_n **coend**

- Critical section statement:

  with r **when** B **do** S

resource r(x): **cobegin**

with r **when** true **do**

$x := x + 1$

||

with r **when** true **do**

$x := x + 1$

coend

- Unable to proove that x is incremented by 2 using the existing axioms.

resource r(x): **cobegin**

  with r **when** true **do**

    $x := x + 1$

||

  with r **when** true **do**

    $x := x + 1$

coend

- The solution: make use of auxiliary variables
  - Auxiliary variable is a variable which is assigned, but never used
  - Removing this variable doesn’t change the program.

resource r(x): **cobegin**

with r **when** true **do**

$x := x + 1$

||

with r **when** true **do**

$x := x + 1$

coend

- If:
  - AV is an auxiliary variable set for a statement S.
  - S' obtained by deleting all assignments to variables in AV.
  - $\{P\}\ S\ \{Q\}$ is true
  - P and Q don't refer to variable any variables from AV.
- Then:
  - $\{P\}\ S'\ \{Q\}$ is also true.

$\{x = 0\}$

begin $y := 0,\ z := 0;$

$\{y = 0 \wedge z = 0 \wedge I(r)\}$

resource $r(x, y, z)$**: cobegin**

$\{y = 0\}$

with r **when** *true* **do**

$\{y = 0 \wedge I(r)\}$

begin $x := x + 1;\ y := 1$ **end**

$\{y = 1 \wedge I(r)\}$

$\{y = 1\}$

||

$\{z = 0\}$

with r **when** *true* **do**

$\{z = 0 \wedge I(r)\}$

begin $x := x + 1;\ z := 1$ **end**

$\{z = 1 \wedge I(r)\}$

$\{z = 1\}$

coend

$\{y = 1 \wedge z = 1 \wedge I(r)\}$

end

$\{x = 2\}$

$I(r) = \{x = y + z\}$

- Each statement has:
  - Pre-condition P
  - Post-condition Q
- Wrote as $\{P\}\ S\ \{Q\}$
- We assume that sequential execution is simple to be proven.
- y and z are auxiliary variables
- $I(r)$ – the invariant for the resource r
  - Remains true at all times outside critical sections for r

$\{\boldsymbol{x = 0}\}$

begin $y := 0,\ z := 0;$

$\{y = 0 \wedge z = 0 \wedge I(r)\}$

resource $r(x, y, z)$**: cobegin**

$\{y = 0\}$

with r **when** $true$ **do**

$\{\boldsymbol{y = 0 \wedge I(r)}\}$

begin $x := x + 1;\ y := 1$ **end**

$\{\boldsymbol{y = 1 \wedge I(r)}\}$

$\{y = 1\}$

||

$\{z = 0\}$

with r **when** $true$ **do**

$\{\boldsymbol{z = 0 \wedge I(r)}\}$

begin $x := x + 1;$ z $:= 1$ **end**

$\{\boldsymbol{z = 1 \wedge I(r)}\}$

$\{z = 1\}$

coend

$\{y = 1 \wedge z = 1 \wedge I(r)\}$

end

$\{x = 2\}$

$\boldsymbol{I(r) = \{x = y + z\}}$

- Each statement has:
  - Pre-condition P
  - Post-condition Q
- Wrote as $\{P\}\ S\ \{Q\}$
- We assume that sequential execution is simple to be proven.
- y and z are auxiliary variables
- $\boldsymbol{I(r)}$ – the invariant for the resource r
  - Remains true at all times outside critical sections for r

$\{x = 0\}$
begin $y := 0,\ z := 0;$
$\{y = 0 \land z = 0 \land I(r)\}$
resource $r(x, y, z)$: **cobegin**
$\{y = 0\}$
with r **when** $true$ **do**
$\{y = 0 \land I(r)\}$
begin $x := x + 1;\ y := 1$ **end**
$\{y = 1 \land I(r)\}$
$\{y = 1\}$
||
$\{z = 0\}$
with r **when** $true$ **do**
$\{z = 0 \land I(r)\}$
begin $x := x + 1;\ z := 1$ **end**
$\{z = 1 \land I(r)\}$
$\{z = 1\}$
coend
$\{y = 1 \land z = 1 \land I(r)\}$
end
$\{x = 2\}$
$I(r) = \{x = y + z\}$

- The critical section axiom:
  - If:
    - $\{I(r) \land P \land B\}\ S\ \{I(r) \land Q\}$
    - $I(r)$ is the invariant from the cobegin statement
    - No variable free in P or Q is changed in another thread
  - Then:
    - $\{P\}$ **with** r **when** B **do** S $\{Q\}$
- For example, set:
  - $P =$ "y = 0"
  - $Q =$ "$y = 1$"
  - $B = true$

$\{x = 0\}$

begin $y := 0,\ z := 0;$

$\{\boldsymbol{y = 0 \wedge z = 0 \wedge I(r)}\}$

resource $r(x, y, z)$**: cobegin**

$\{y = 0\}$

with r **when** $true$ **do**

$\{y = 0 \wedge I(r)\}$

begin $x := x + 1;\ y := 1$ **end**

$\{y = 1 \wedge I(r)\}$

$\{y = 1\}$

||

$\{z = 0\}$

with r **when** $true$ **do**

$\{z = 0 \wedge I(r)\}$

begin $x := x + 1;\ z := 1$ **end**

$\{z = 1 \wedge I(r)\}$

$\{z = 1\}$

coend

$\{\boldsymbol{y = 1 \wedge z = 1 \wedge I(r)}\}$

end

$\{x = 2\}$

$I(r) = \{x = y + z\}$

- The parallel execution axiom:
  - If:
    - $\{P_1\}\ S_1\ \{Q_1\}\ \ldots\ \{P_n\}\ S_n\ \{Q_n\}$
    - No variable free in P_i or Q_i is changed in S_j $(i \neq j)$
    - All variables in $I(r)$ belong to resource r
  - Then:
    - $\{P_1 \wedge \ldots \wedge P_n \wedge I(r)\}$ **resource** r**: cobegin** $S_1 // \ldots // S_n$ **coend** $\{Q_1 \wedge \ldots \wedge Q_n \wedge I(r)\}$
- For example, set:
  - $P_1 = "y = 0"$
  - $P_2 = "z = 0"$
  - $Q_1 = "y = 1"$
  - $Q_2 = "z = 1"$

$\{x = 0\}$

begin $y := 0,\ z := 0;$

$\{y = 0 \wedge z = 0 \wedge I(r)\}$

resource $r(x, y, z)$: **cobegin**

$\{y = 0\}$

with r **when** *true* **do**

$\{y = 0 \wedge I(r)\}$

begin $x := x + 1;\ y := 1$ **end**

$\{y = 1 \wedge I(r)\}$

$\{y = 1\}$

$||$

$\{z = 0\}$

with r **when** *true* **do**

$\{z = 0 \wedge I(r)\}$

begin $x := x + 1;\ z := 1$ **end**

$\{z = 1 \wedge I(r)\}$

$\{z = 1\}$

coend

$\{y = 1 \wedge z = 1 \wedge I(r)\}$

end

$\{\boldsymbol{x = 2}\}$

$I(r) = \{x = y + z\}$

- Using the invariant $I(r)$, we have the result:

$\boldsymbol{x = 2}$