Valigator Examples
Thibaud Hottelier

June 20, 2008

1 Simple Counter

Program code:

```c
assume(x < 10);
while(x < 10) {
    x = x + 1
};
x = x + 1;
assert(x = 11)
```

Valigator output when invoked with “valigator –cvc3 examples/counter”:

```
Starting analysis...
Info: Program is linear
Info: 3 proof obligation(s) generated
Running CVC3 for proving...

**********************************************************************
* Result: YES (All proof obligations proved!) *
**********************************************************************
```

The invariant inferred by Aligator is

\[(x_0 < 10 \Rightarrow x \leq 10) \land (x_0 \geq 10 \Rightarrow x = x_0)\]

2 Adder

Program code:

```c
assume(b >= 0);
result = a;
counter = b;
while(counter > 0) {
    counter = counter - 1;
    result = result + 1
};
assert(result = a + b)
```

Valigator output when invoked with “valigator –cvc3 examples/adder”:
Starting analysis...
Info: Program is linear
Info: 3 proof obligation(s) generated
Running CVC3 for proving...

******************************************************************************
* Result: YES (All proof obligations proved!) *
******************************************************************************

The invariant inferred by Aligator is

\[ \text{counter} + \text{result} = a + b \land (b \leq 0 \Rightarrow \text{counter} = b \land \text{result} = a) \land (b > 0 \Rightarrow \text{counter} \geq 0) \]

3 Cousot 77

Program code:

```c
x = a;
y = b;
while((x > y) | (y > x)) {
    x = x + 1;
y = y - 1
};
assert(x = y & 2*x = a + b)
```

Valigator output when invoked with “valigator –cvc3 examples/cousot77”:

Starting analysis...
Info: Program is linear
Info: 3 proof obligation(s) generated
Running CVC3 for proving...

******************************************************************************
* Result: YES (All proof obligations proved!) *
******************************************************************************

The invariant inferred by Aligator is

\[ x + y = a + b \land ((a < b \lor a > b \lor (a = x \land b = y)) \land (x < y \lor x > y \lor (a + b = 2 \times x \land a + b = 2 \times y))) \lor (a = b \land a = x \land b = y) \]

4 Fibonacci Even

Program code:

```c
assume(cnt > 0);
f0 = 0;
f1 = 1;
while(cnt > 0) {
    tmp = f1;
f1 = f1 + f0;
f0 = tmp;
}
\[
cnt = cnt - 1
\]

\[
assert(!(f0 \% 2 = 0 \& f1 \% 2 = 0))
\]

Valigator output when invoked with “valigator examples/fiboEven”:

Starting analysis...
Info: Program is linear
Info: 3 proof obligation(s) generated
Running STP for proving...

*****************************************************************************
* Result: YES  (All proof obligations proved!)  *
*****************************************************************************

The invariant infered by Aligator is

\[
f0^2 + 2 \cdot f0^3 \cdot f1 + f1^4 = 1 + f0^2 \cdot f1^2 + 2 \cdot f0 \cdot f1^3 \land (cnt0 > 0 \Rightarrow cnt \geq 0) \\
\land (cnt0 \leq 0 \Rightarrow f1 = 1 \land f0 = 0 \land tmp = tmp0 \land cnt = cnt0)
\]

5 Commuting branches

Program code:

\[
x = 0;
\]
\[
c = 0;
\]
\[
a = 0;
\]
\[
while(x < 10) {
    if (x \% 2 = 0) {
        c = c + x
    } else {
        a = a + x
    }
};
\]
\[
x = x + 1
\]
\[
assume(a \geq 0 \& a \leq 63 \& c \geq 0 \& c \leq 63);
\]
\[
assert(a + c = 45)
\]

Valigator output when invoked with “valigator examples/commutable”:

Starting analysis...
Info: Program is linear
Info: 3 proof obligation(s) generated
Running STP for proving...

*****************************************************************************
* Result: YES  (All proof obligations proved!)  *
*****************************************************************************

The invariant infered by Aligator is

\[
2 \cdot a + 2 \cdot c + x = x^2 \land x \leq 10
\]
6 10\textsuperscript{th} Fibonacci number

Program code:

```c
int cnt = 10;
int f0 = 0;
int f1 = 1;
while(cnt > 0) {
    int tmp = f1;
    f1 = f1 + f0;
    f0 = tmp;
    cnt = cnt - 1
}
assert(f1 = 89)
```

Valigator output when invoked with “valigator examples/fibo10”:

```
Starting analysis...
Info: Program is linear
Info: 3 proof obligation(s) generated
Running STP for proving...

******************************************************************************
* Result: NO (One or more assertion disproved) *
******************************************************************************
```

The invariant inferred by Aligator is

\[ f_0^4 + 2 \cdot f_0^3 \cdot f_1 + f_1^4 = 1 + f_0^2 \cdot f_1^2 + 2 \cdot f_0 \cdot f_1^3 \land cnt \geq 0 \]

This example fails because the invariant is not strong enough. It does not relate in any way \(cnt\) and \(f_1\) or \(f_0\). The invariant we would need to infer in order to prove this example is the closed formula of the Fibonacci numbers, which involves exponentials and is therefore completely out of reach.

7 Unsound Invariant

Program code:

```c
assume(a >= 0);
x = a;
cnt = 1;
while(cnt > 0) {
    cnt = cnt - 1;
    x = (2*(x/2)) + 1
}
assert(x = a + 2)
```

Valigator output when invoked with “valigator examples/btest”:

```
Starting analysis...
Info: Program is linear
```
Info: 3 proof obligation(s) generated
Running STP for proving...

**********************************************************************
* Result: UNKNOWN         (Aligator generated an invalid invariant) *
**********************************************************************

The incorrect invariant inferred by Aligator is

\[ cnt + x = 1 + a \land cnt \geq 0 \]