Loop Bound Analysis

Pritom Rajkhowa

May 2019

1 Motivating Examples

1.1 Example 1

However, we want to highlight the major weakness of existing approaches with the help of the following examples of C code snippet

P_1	P_2

<pre>int x , C;
while(x < C)
{
   x = x + 1;
}</pre>
	<pre>int x ,y,C;
	while(x + y < C)
	{
	   x = x + 1;
	   y = y + 1;
	}</pre>

The potential-based techniques and amortized analysis base tool C^4B can easily find the bound for the program P_1 where C^4B return the bound $1.00|[0, C]|$ for the program P_1. But all the tools including C^4B failed to find the bound for the program P_2. VIAP can find the bound of both programs $|[0, C-x]|$ and $|[0,(C-x-y)/2]|$ for P_1 and P_2 respectively. To illustrate how VIAP works, consider the program P_2. Lin's [1] translation of the above program P generates the following a set of axioms $\Pi_P^{\vec{X}}$ after some simple simplifications where $\vec{X} = \{x, y, C\}$

$$\begin{aligned}
&C_1 = C, x_1 = x_3(n), y_1 = y_3(n),\\
&x_3(0) = x, y_3(0) = y,\\
&x_3(n+1) = x(n) + 1, y_3(n+1) = y_3(n) + 1,\\
&\neg(x_3(N) + y_3(N) < C),\\
&\forall n. n < N \rightarrow (x_3(n) + y_3(n) < C)
\end{aligned}$$

where x_1, y_1 and C_1 denote the output values of a, b, z, x and y, respectively, $x_6(n)$ and $y_6(n)$ the values of x and y during the nth iteration of the loop, respectively. Also N is a natural number constant, and the last two axioms say that it is exactly the number of iterations the loop executes before exiting. Our recurrence solving tool recSolve(RS) can find the closed-form solutions of $x_3(n)$ and $y_3(n)$, which yields the closed-form solution $x_3(n) = n - x$ and $y_3(n) = n - y$ respectively. Those can be used to simplify the recurrence for $y_3(n)$ into following set of axoims after eliminates recurrence relation for $x_3()$, and $y_3()$

$$\begin{aligned}
&x_1 = N + x, y_1 = N + y, C_1 = C,\\
&(2 * N + x + y \geq C)\\
&\forall n. n < N \rightarrow (2 * n + x + y < C)
\end{aligned}$$

The system tried to derive using our algorithm derived $N = (C - x - y)/2$, then it tried to prove using SMT solver. If it is able to successfully prove that, then it successfully derived bound $|[0, (C - x - y)/2]|$.

References

[1] F. Lin, "A formalization of programs in first-order logic with a discrete linear order," *Artificial Intelligence*, vol. 235, pp. 1 – 25, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000437021630011X