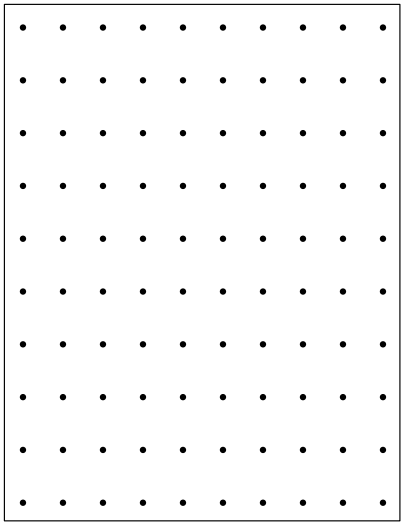
A General Approach for Tight Timing Predictions of Non-Rectangular Loops

Christopher A. Healy
Robert A. van Engelen
David B. Whalley

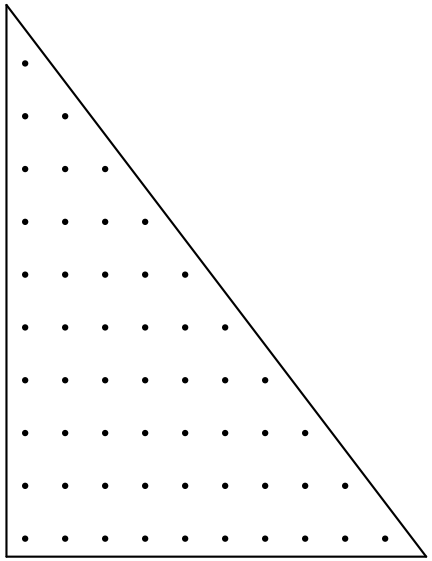
Florida State University

Rectangular versus Non-Rectangular Loop Nests

```
for (i = 0; i < 10; i++)
   for (j = 0; j < 10; j++)
```



```
for (i = 0; i < 10; i++)
   for (j = 0; j < i; j++)
```



Goals

- Automatically calculate an accurate number of iterations for non-rectangular loop nests.
- Use a general approach that is not limited to nesting depth or dependences between index variables among loops in the nest.
- Integrate with an existing timing analyzer to obtain tight WCET and BCET of every loop.

Formulating the Number of Iterations as a Summation

- For a simple loop,

```
for (i = a; i <= b; i++) ...
```

  we define the number of iterations as follows:

$$I = \sum_{i=a}^{b} 1 = \begin{cases} b - a + 1 & \text{if } a \le b \\ 0 & \textit{otherwise} \end{cases}$$

- Constraint on bounds is necessary due to zero-trip loops.

A Partially Zero-Trip Loop

```
for (i=1; i<8; i++)
   for (j=i; j<3; j++)
```

$$I = \sum_{i=1}^{7} \begin{cases} 3 - i & \text{if } i \le 2 \\ 0 & \textit{otherwise} \end{cases}$$

- This summation equals 3, but a naive calculation would result in –7.

Dealing with Nonunit Strides

```
for (i = a; i <= b; i += s)...
```

- Summations involving nonunit strides are converted to uniform summations according to:

$$I = \sum_{i=a}^{b,s} e = \sum_{i=0}^{\lfloor (b-a)/s \rfloor} e\,[i \leftarrow si + a]$$

— All free occurrences of i are replaced by $si + a$.

— The resulting summation contains floor expressions, which can be rewritten as modulo operations.

Detecting the Absence of a Partially Zero Trip Loop

```
for (i = 0; i < 10; i++)
   for (j = i; j < 11; j++)
      for (k = i-3; k < j+8; k++)
```

we expand the inital value and limit:

$$i - 3 = [0 - 3 \,..\, 9 - 3] = [-3 \,..\, 6]$$
$$j + 8 = [i + 8 \,..\, 10 + 8] = [8 \,..\, 18]$$

- The ranges do not overlap, so the loop nest is not partially zero trip.

Conclusion

- Developed a general approach to count loop iterations as a nested summation.
- Timing analyzer formulates summation expression, evaluates the sum, and computes average number of iterations to compute tight WCET/BCET.
- Currently working on arbitrary nonunit strides within a loop nest.
- On-line demo available:
  `http://www.cs.fsu.edu/~engelen/iternum.cgi`