Checking non-interference in SPMD (CUDA) programs

Stavros Tripakis, Christos Sergiou

UC Berkeley

Roberto Lublinerman

Penn State

What: method and tool to check non-interference in CUDA programs

Why: non-interference => absence of races => determinism

How: parse program, generate formulas, submit to SMT solver

```
__global__ void arrayRevCoalGPU(
    float *Out,    float *In,    int N
){
  __shared__ float Local[BLOCKSIZE];

  int i = blockIdx.x * blockDim.x + threadIdx.x;
  int j = (gridDim.x-1-blockIdx.x)*blockDim.x + ...;
  int k = blockDim.x-1-threadIdx.x;

  Local[threadIdx.x] = In[i];
  __syncthreads();
  Out[j] = Local[k];
}
```

CUDA program

```
(define t1:: int)
(define t2:: int)
(define b1:: int)
(define b2:: int)
(assert (or (/= t1 t2) (/= b1 b2)))
(assert (>= t1 0)) (assert (>= t2 0))
(assert (>= b1 0)) (assert (>= b2 0))
(assert (< t1 256))
(assert (< t2 256))
(assert (< b1 256))
(assert (< b2 256))
(assert (= (+ (* b1 256) t1) (+ (* b2
                    256) t2)))
(check)
```

Generated formula