SMART: **S**imultaneous **M**ultithreading **A**pplied to **R**eal **T**ime

Sims Osborne and James H. Anderson

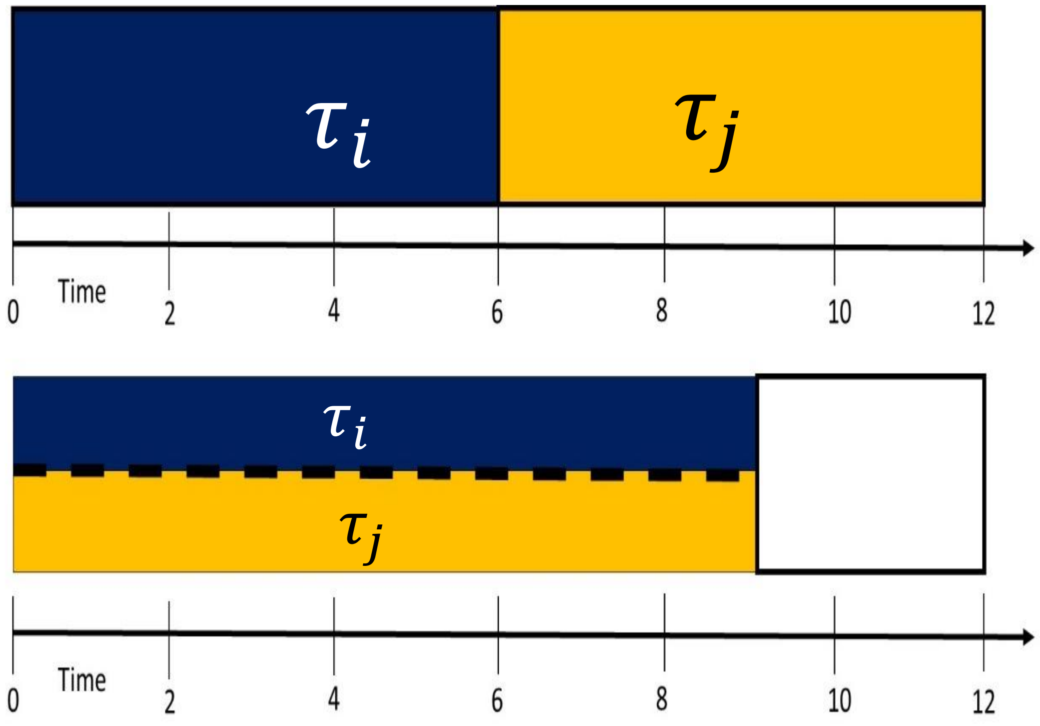
THE UNIVERSITY *of* NORTH CAROLINA *at* CHAPEL HILL

The SMART model: $\tau_i = \left(T_i, C_i, \left(r_{i:j}\ \forall\ \tau_j\right)\right)$

$r_{i:j}$: How fast does τ_i execute with τ_j?

$$r_{i:j} = \frac{C_i\ (executing\ without\ SMT)}{C_i\ (co\text{-}scheduled\ with\ \tau_j)}$$

$r_{i:j}$	Interpretation given 2 threads
<0.5	τ_i should not execute as thread
>.5	τ_i may benefit from threading
=1	τ_i has same cost as thread



Example: $r_{i:j} = \frac{6}{9} = \frac{2}{3}$

Measured benchmark \ Interfering benchmark	adpcm_dec	adpcm_enc	ammunition	cjpeg_transupp	cjpeg_wrbmp	dijkstra	epic	fmref	gsm_dec	gsm_enc	h264_dec	huff_dec	mpeg2	ndes	rijndael_dec	rijndael_enc	statemate	susan
adpcm_dec	0.90	1.01	0.93	1.00	1.00	0.92	1.00	0.90	1.00	1.00	0.91	0.93	1.00	0.92	0.93	1.00	0.94	0.93
adpcm_enc	0.92	1.00	1.00	1.00	1.00	0.99	0.94	0.90	0.91	1.00	0.99	1.00	0.92	1.00	0.93	0.93	1.00	1.00
ammunition	0.68	0.67	0.68	0.66	0.70	0.69	1.00	0.65	0.66	1.00	1.00	0.69	0.67	0.69	0.69	0.71	0.72	0.70
cjpeg_transupp	0.68	0.66	0.69	1.00	0.65	0.64	0.72	1.00	0.62	0.65	1.00	0.64	0.64	0.67	1.00	0.68	0.67	0.62
cjpeg_wrbmp	1.00	0.64	0.67	0.55	0.58	0.60	0.99	1.00	1.00	0.61	0.61	0.59	0.61	0.64	0.62	0.62	1.00	0.61
dijkstra	0.72	0.71	1.00	1.00	1.00	1.00	0.76	0.68	0.68	0.70	0.72	0.73	0.70	0.72	0.73	0.73	0.69	0.72
epic	0.54	0.53	1.00	0.54	1.00	1.00	0.56	0.56	0.55	0.54	0.57	0.54	0.51	0.54	0.55	0.55	0.54	0.54
fmref	0.71	0.72	0.73	0.97	0.71	0.70	0.74	0.67	0.70	0.70	0.69	0.69	0.69	0.71	0.71	0.71	0.74	0.70
gsm_dec	0.65	0.64	0.65	0.61	0.66	0.64	0.68	0.61	0.61	0.63	0.62	0.65	0.63	0.65	0.64	1.00	0.66	0.62
gsm_enc	0.59	0.58	0.61	0.59	0.99	0.62	0.64	0.56	0.57	0.59	0.60	0.61	0.58	0.61	0.61	0.62	0.63	0.59
h264_dec	1.04	0.90	0.90	1.04	0.87	0.84	0.95	0.86	0.80	1.00	0.85	0.88	0.86	0.89	0.84	0.84	0.88	0.84
huff_enc	0.73	0.71	0.73	0.65	1.00	0.65	1.00	1.00	1.00	1.00	0.68	0.69	0.65	0.70	0.70	1.00	0.70	0.68
mpeg2	0.71	0.71	0.73	0.69	0.72	0.70	0.75	0.69	0.68	0.70	0.99	0.99	0.68	0.72	0.71	0.73	0.70	0.70
ndes	1.00	0.57	0.60	0.57	0.61	0.98	0.60	1.01	0.56	0.53	0.59	1.00	0.52	0.60	0.59	0.60	0.62	0.60
rijndael_dec	1.00	0.58	0.57	0.62	0.64	0.65	0.66	0.63	0.60	0.62	0.64	0.65	0.62	1.00	0.62	0.60	0.63	1.00
rijndael_enc	0.56	0.57	0.58	0.62	0.63	0.63	0.65	1.00	0.57	0.60	0.63	0.63	0.60	0.61	0.59	0.59	1.00	1.00
statemate	0.62	0.62	0.65	0.51	0.51	0.56	0.73	1.00	0.64	0.59	0.58	0.55	1.00	0.58	0.65	0.65	0.50	0.57
susan	0.65	0.60	0.63	0.58	1.00	0.58	0.69	0.61	0.58	0.93	0.58	1.00	0.60	0.61	0.63	0.64	0.66	1.00

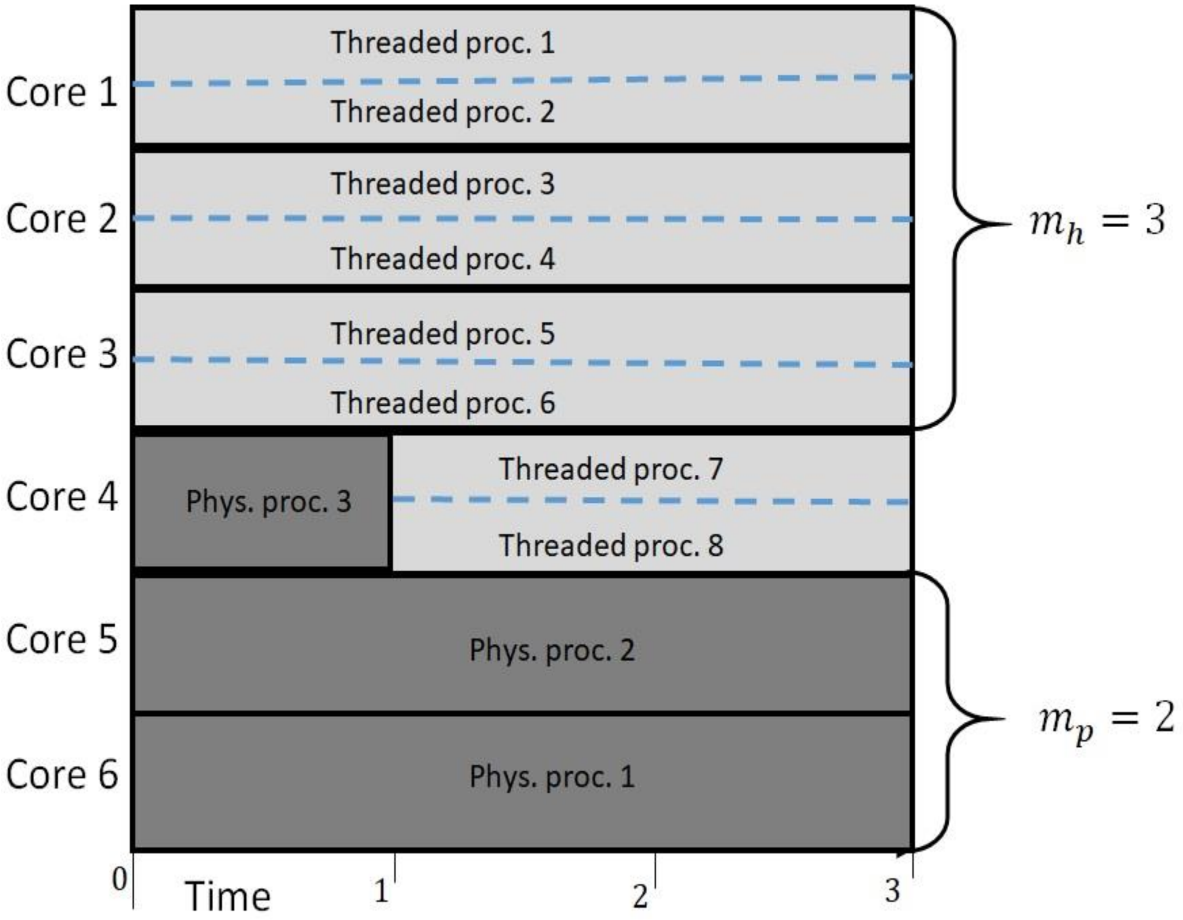
Key

- $r_{i:j} < 0.65$
- $0.65 \le r_{i:j} < 0.80$
- $0.80 \le r_{i:j}$

$r_{i:j}$ for tasks from TACLeBench on Intel Xeon Silver 4110 2.1 GHz (Q3 '17)

Method 1: Partition Tasks

- Every task gets a physical and a threaded cost: $\tau_i = (T_i, C_i^p, C_i^h)$.
- Tasks are threaded if
  - $\frac{C_i^h}{T_i} \leq 1$ and
  - $\frac{C_i^p}{C_i^h} \geq \frac{1}{2}$.
- Task types scheduled separately.



Sample Results

Scheduling Method	Effective Utilization	Reduction
Traditional	2.125	NA
Naïve physical/ threaded partition	1.875	12 %
Advanced physical/ threaded partition	1.833	14%
Individual Pairings	1.782	16%
Ideal	1.742	18%

Buy 8 cores, get 1 free!

On Poster

- Further model details.
- How to partition tasks.
- Feasibility conditions.

Future Work

- Large scale schedulability study.
- Higher thread counts.
- Mixed criticality applications.