

根据 CS:APP 上的介绍, core-i7 的寄存器 L1 d-cache 是 64 组 8 路组相联 (P577), 则 32KB 的 L1 缓存, 一组中一行的大小为:

$$\text{linesize} = 32\text{KB} \div 64 \div 8 = 2^5 \div 2^6 \div 2^3 = 2^6 = 64 \text{ byte} = 8 \text{ int (64 位机)}$$

∴ 当访问大小为  $1024 \times 1024$  的矩阵中的  $[i][j]$  元素时, 其缓存所在的 Cache 组为

$$\text{set} = \left[ (1024 \times i + j) / 8 \right] \% 64$$

↳ 取 ~~set~~ floor

∴ 同理访问大小为  $1025 \times 1025$  的矩阵中的  $[i][j]$  元素时, 其缓存所在的 Cache 组为

$$\text{set}' = \left[ (1025 \times i + j) / 8 \right] \% 64$$

↳ 取 ~~set~~ floor

矩阵转置时, 访问  $[i][j]$  的同时也要访问  $[j][i]$ , 根据上面两个公式, 可以分别计算  $i=0$  时 cache 组 (set) 的访问情况, 如下表

	1024 × 1024		1025 × 1025	
$i=0, j=0$	$[0][0]$	set=0	$[0][0]$	set=0
	$[0][1024]$	set=0	$[0][1025]$	set'=0
$i=0, j=1$	$[0][1]$	set=0	$[0][1]$	set'=0
	$[1][0]$	set=0	$[1][0]$	set'=0
	⋮		⋮	
$i=0, j=8$	$[0][8]$	set=1	$[0][8]$	set'=1
	$[8][0]$	set=0	$[8][0]$	set' = $\lfloor 1025 \times 8 / 8 \rfloor \% 64 = 1$
$i=0, j=9$	$[0][9]$	set=1	$[0][9]$	set'=1
	$[9][0]$	set=0	$[9][0]$	set' = $\lfloor 1025 \times 9 / 8 \rfloor \% 64 = 1$

1831604

张尹嘉

可以发现当矩阵大小是  $1024 \times 1024$  时, 访问对称位置时对应的 Cacheset 均为 0, 则当 0 号 Cacheset 中 64 个 int 会满时, 就会引起与内存的读写。而当矩阵大小是  $1025 \times 1025$  时, 当  $j \geq 8$  时就会访问到 1 号 Cacheset, 后续发生 Cacheset 会满的次数会大大减少。

