

Rainmaker's Notebook

『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

Home

Archives

About

SiteXC

SC19: 年轻人的第一次公费旅游

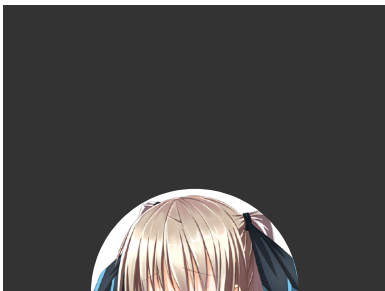
这是一篇咕咕咕了一个月的游记。

去年老板 carry 中了一篇 SC18 论文，去开了年轻人的第一次学术会议。今年因为各种原因，没有合适的工作投 SC19，只好去公费旅游了。原本没中论文老板是不会掏钱的，但是我五月的时候本该去巴西开会因为签证问题没去成，权当给我补了一次。说是公费旅游其实也不准确，应该说是公费杨白劳：去年机缘巧合跟 Rebecca，今年 Student Cluster Competition 的 chair 聊了一下，被她拉到了 SCC19 组委会里打杂，某种意义上也算在 SC19 的组委会里了。然而组委会是义务工作，门票钱都不能免，所以就是公费去做杨白劳了。

SC19 在丹佛开，大概是第三次还是第四次在丹佛开了。我 17 号的飞机去丹佛。从丹佛机场出来，放眼望去就是荒草枯黄平原上矗立的积雪的科罗拉多山脉。山顶的积雪在阳光的照耀下倒是让人觉得心旷神怡。从丹佛机场到丹佛市区有轨道交通，比起达拉斯来真是好得多了。轨道交通坐到 Union Station，距离会议中心只有一公里，我决定走过去，这样可以看看丹佛的市区。丹佛这种明日黄花的美国中西部老工业城市，网上人称 middle of nowhere，走起来看的确有一点这样的感觉。会议中心周围倒是人多，多得让我有一种回国的感觉。

到了会场，组委会暂时并没有什么事情需要做，我就先把 console 线拿给了哈利橙。起因是我出发前两天，哈利橙在 TUNA 群里问有没有人要来 SC19，求捎一根交换机 console 线，我就正好找江大借了一根带过去。这次来 SCC，见到了两个在 TUNA 群里活跃的面孔：哈利橙和丽狼。想想人家大三已经做了什么，我大三在混什么日子，不得不承认 Top 2 大学的学生和五本大学的学生之前有很大差距。上交队里我也有个不怎么熟的旧相识：他们的带队老师廖秋承比我大一级，ASC16 的时候我们打过照面，然而后来没有联系。廖现在在上交超算中心干活，这几天陆陆续续给我讲了不少东西。其中包括他们测了 AMD Rome CPU 各方面表现非常好，以及 Intel Xeon 自 Skylake 以后的同型号不同批次的主频和性能波动问题，听得我只想喊 AMD Yes。

去年到会场的第一晚，蹭了广州超算中心卢主任请客的一顿大餐，今年就没有这个享受了：卢主任和杜总工都因为签证问题来不了。与此同时，会场里原本华为预定的一大块场地也没有人来布展，直接空出来了。这倒是让我有点失望：原本还想去问问他们的 ARM 芯片的情况。



Rainmaker's Notebook

『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

[Home](#)

[Archives](#)

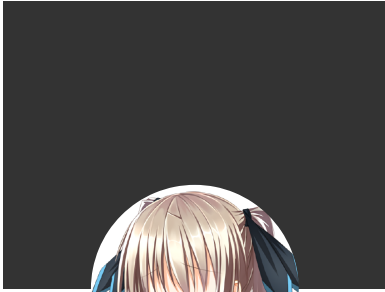
[About](#)

[SiteXC](#)

18号是 SC19 厂商布置展台的最后一天，也是 SCC19 开赛的第一天。SCC 的赛程较 ASC 更长，第一天只跑基准测试。今年除了 HPL 和 HPCG 两个保留项目以外，还加进来一个分数几乎可以忽略不计鼓励大家志在参与的 IO-500。我早上和下午基本都在 SCC 会场看场子，其实也没有什么事情要做，基本都是自己在笔记本上干活。今年的 HPL 又是南洋理工拿了第一，这毫不令我感到意外：他们跟 NV 有深度的合作，连 HPL/HPCG 的 binary 都和其他队伍的不一样，是优化得更好一点的新版本。下午 5 点半基准测试结束以后，有两个小时的休息时间，正好对着开幕式自助餐。

晚上 7 点 SC19 展览正式开始。Intel 正式公开了他们的 oneAPI 框架，说是一个编程框架包办 CPU, GPU, NPU (神经网络处理器，基本上就是只能做矩阵乘法的芯片)，和 FPGA。然而我感觉这只是个 PPT 产品，强行把编程特点不一样的硬件用一个编程框架套起来，只能是胶水而已，最后需要性能的时候还是要回落到 C/OpenCL code。某种意义上，AMD 倒是通过便乘 CUDA，在 GPU 的赛道上追突 NV 绿色高级车了：AMD 的 HIP 框架的命名几乎是照抄 CUDA 的，甚至连内部培训用的编程手册里的图都是直接截了 CUDA 资料的图。因此，AMD 可以直接将大部分不涉及硬件专用函数的 CUDA code 转化为 HIP code 然后编译。从 AMD 给出的性能数据来看，他们已经移植好的那几个程序在 MI 50/60 上的性能已经可以和 P100/V100 上的媲美了。就算去掉厂商的有色眼镜，这大概也表明现有的 CUDA code 移植到 AMD 平台后，性能能接近 NV 平台了（均以两者的顶级计算卡来算）。牙膏厂的 oneAPI 和 Xe 显卡到现在还没拿出实际的产品来，我是很怀疑 2021 年他们准备怎么给 DOE 交 E 级机的货了。

我在展区里路过 NEC 展台的时候惊奇地发现他们那个 Vector Engine 今天不仅带了展示机来，还允许大家上手试验。我问了一下，他们允许我自己写个 code 然后跑。NEC 号称他们的编译器能处理好向量化问题，那我自然是要试一下了。我立即回去用最快速度从以前的代码里截出来了两个小程序：一个是 3-axis stencil kernel，另一个是 CSR SpMV，都是串行的程序，然后拿去 NEC 的展台测试。选这两个程序的原因在于：(1) 这两个程序并行化以后都是 memory bandwidth bound 的，而 VE 的卖点之一是 6 块 HBM2 堆起来高达 1.2TB/s 的总内存带宽；(2) 这两个程序的向量化都是易做难精，真要向量化性能好少不了针对性的优化。结果很不幸，VE 当场翻车了：NEC 编译器编译出来到 VE 上跑的程序似乎是没有并行化的，只有向量化，但是还不如在 Xeon Gold 61xx 的 Host CPU 上用 gcc 4.8.5 这种老货色编译出来的程序跑得快！更讽刺的是，stencil kernel 还是 NEC 宣传的能达到很好性能的卖点之一。NEC 大概也没有想到遇上我这么一个见到锤子还真抡起来砸场子的人。顺带吐槽一点，NEC 在会场上发的资料，简单介绍怎么用他们的编译器的，大概也是没打算让米英鬼畜们看懂的：和 xxx is good 对应的不是 xxx is bad，而是 xxx is ng (not good, 日式英语)



Rainmaker's Notebook

『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

Home

Archives

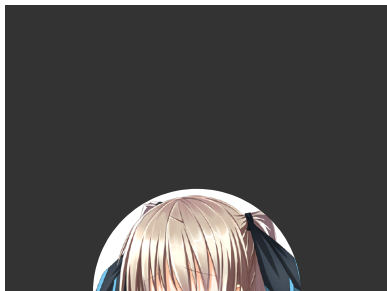
About

SiteXC

```

[handson@handson03 ~]# ncc stencil_3axis_CSR.c
stencil_kernel finished, used time = 0.103931 (s), 0.108992 GPoint/s
stencil_kernel finished, used time = 0.150055 (s), 0.111807 GPoint/s
[handson@handson03 mkurisu]$
[handson@handson03 mkurisu]$
[handson@handson03 mkurisu]$ ./stencil_3axis_SpMV.ncc 256 256 256
-bash: ./stencil_3axis_SpMV.ncc: No such file or directory
[handson@handson03 mkurisu]$ ls
stencil_3axis.c stencil_3axis_CSR.c stencil_3axis_CSR gcc stencil_3axis gcc stencil_3axis.ncc
[handson@handson03 mkurisu]$ ncc stencil_3axis_CSR.c -o stencil_3axis_CSR.ncc
"stencil_3axis_CSR.c", line 143: warning: last line of file ends without a
newline
)
ncc: vec( 103): stencil_3axis_CSR.c, line 51: Unvectorized loop.
ncc: vec( 101): stencil_3axis_CSR.c, line 106: Vectorized loop.
ncc: vec( 101): stencil_3axis_CSR.c, line 109: Vectorized loop.
ncc: vec( 103): stencil_3axis_CSR.c, line 129: Unvectorized loop.
[handson@handson03 mkurisu]$ ./stencil_3axis_SpMV.ncc 256 256 256
-bash: ./stencil_3axis_SpMV.ncc: No such file or directory
[handson@handson03 mkurisu]$ ./stencil_3axis_CSR.ncc 256 256 256
Test parameters:
  Domain size x,y,z = 256 * 256 * 256
  Test repeats      = 5
CSR SpMV 0: 0.180 GFlops
^Ckilled
[handson@handson03 mkurisu]$ ./stencil_3axis_CSR.ncc 128 128 128
Test parameters:
  Domain size x,y,z = 128 * 128 * 128
  Test repeats      = 5
CSR SpMV 0: 0.187 GFlops
CSR SpMV 1: 0.187 GFlops
CSR SpMV 2: 0.187 GFlops
CSR SpMV 3: 0.187 GFlops
CSR SpMV 4: 0.187 GFlops
[handson@handson03 mkurisu]$ ./stencil_3axis_CSR.ncc 128 128 128
Test parameters:
  Domain size x,y,z = 128 * 128 * 128
  Test repeats      = 5
CSR SpMV 0: 1.631 GFlops
CSR SpMV 1: 1.672 GFlops
CSR SpMV 2: 1.669 GFlops
CSR SpMV 3: 1.648 GFlops
CSR SpMV 4: 1.687 GFlops
[handson@handson03 mkurisu]$

```



Rainmaker's Notebook

『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

Home

Archives

About

SiteXC

```

handson@handson03:~/mkurisu
[handson@handson03 mkurisu]$ gcc stencil_3axis.c -o stencil_3axis.ncc
"stencil_3axis.c", line 152: warning: last line of file ends without a newline

ncc: vec( 101): stencil_3axis.c, line 42: Vectorized loop.
ncc: vec( 101): stencil_3axis.c, line 60: Vectorized loop.
ncc: vec( 102): stencil_3axis.c, line 97: Partially vectorized loop.
ncc: vec( 101): stencil_3axis.c, line 101: Vectorized loop.
ncc: vec( 103): stencil_3axis.c, line 125: Unvectorized loop.
[handson@handson03 mkurisu]$ ./stencil_3axis.gcc 256 256 256
Test parameters:
  Domain size x,y,z = 256 * 256 * 256
  Test repeats      = 5

Test target: 3-axis stencil, radius = 6 (37 points stencil)

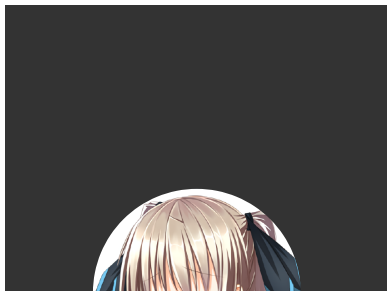
Stencil kernel finished, used time = 0.151661 (s), 0.110623 GPoint/s
Stencil kernel finished, used time = 0.151117 (s), 0.111021 GPoint/s
Stencil kernel finished, used time = 0.153228 (s), 0.109492 GPoint/s
Stencil kernel finished, used time = 0.151006 (s), 0.111103 GPoint/s
Stencil kernel finished, used time = 0.152854 (s), 0.109760 GPoint/s
[handson@handson03 mkurisu]$ ./stencil_3axis.ncc 256 256 256
Test parameters:
  Domain size x,y,z = 256 * 256 * 256
  Test repeats      = 5

Test target: 3-axis stencil, radius = 6 (37 points stencil)

Stencil kernel finished, used time = 0.555258 (s), 0.030215 GPoint/s
Stencil kernel finished, used time = 0.555158 (s), 0.030221 GPoint/s
Stencil kernel finished, used time = 0.555065 (s), 0.030226 GPoint/s
Stencil kernel finished, used time = 0.555085 (s), 0.030225 GPoint/s
Stencil kernel finished, used time = 0.555065 (s), 0.030226 GPoint/s
[handson@handson03 mkurisu]$

```

路过富士通展台的时候他们正好在搞某个类似开幕式的东西，打开了一桶日本清酒，分发给在场人员。我拿了一杯，合起来感觉酒劲还不小。我原本以为富士通大部分都是日本人，没想到还有几个老外。然而富士通对他们的 A64FX 守口如瓶，什么都不说。我对 A64FX 颇感兴趣，虽然某种意义上 A64FX 很像 Xeon Phi 的 ARM 内存加强版，但是松岗聪对它赞誉有加，不得不让我感兴趣。强劲的 CPU 核心加上足够大的内存带宽，很可能跑程序不比同等内存带宽下的 GPU 差，毕竟现在大部分程序都是 memory BW bound，没有多少真的需要那么高的浮点性能的。哎，要是能有机会玩一下 A64FX 就好了.....



Rainmaker's Notebook

『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

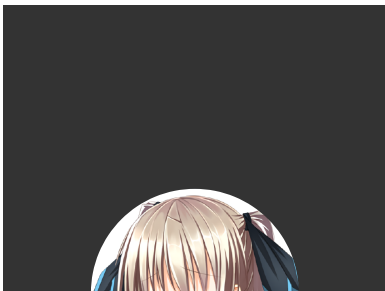
Home

Archives

About

SiteXC





Rainmaker's Notebook

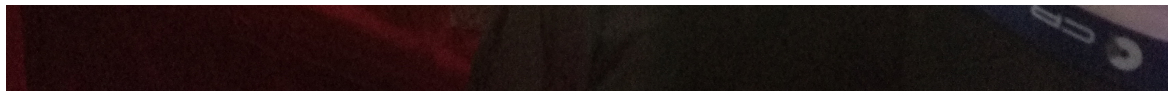
『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

[Home](#)

[Archives](#)

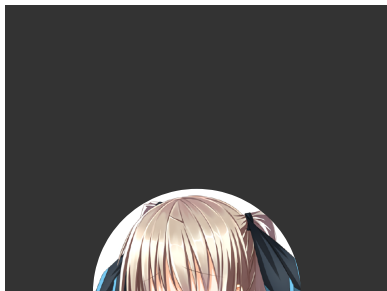
[About](#)

[SiteXC](#)



19号早上起了个大早，因为早上7点是组委会预定的第一次断电挑战。断电挑战非常反映现实了，LBNL的Cori今年下半年就因为加州山火和供电的问题停过。SCC的赛制比起ASC更折腾人的是晚上机器不断电可以继续跑，因此每个队伍都要有人守夜。听说去年是凌晨2点断的电，这就非常折磨人了，不仅折磨队员，还折磨组委会成员。好在组委会今年定的两个时间都比较人道（早上7点和晚上8点）。下午我去听了三篇论文。一篇是讲做distributed memory matrix multiplication的，理论通信量比起SOTA的2.5D算法还要低，我没想到这个题目还能百尺竿头更进一步。特别是，Demmel和其合作者已经证明了2.5D算法的通信量与理论下限是同阶的。顺带一提，我之前写论文的时候试过2.5D和3D算法。3D算法虽然理论通信量比2.5D和2D的小，但是在进程数少的时候（比如 $P=4^3=64$ ），2D可能比3D/2.5D更快。原因在于2D可以只做点对点通信跑满带宽，而2.5D/3D的集合通信真要算起来对带宽的利用率很低，中间同步以及其他的overhead比较大。这就是我之前一篇论文的一个优化切入点。第二个是中科院计算所的人的论文，讲做FFT优化的，讲的那个人讲得实在不行。第三篇论文是Dongarra那边的人做的，一个新的线性代数包，旨在取代LAPACK和ScaLAPACK，受到了很多的关注。听完论文以后我见到了James Demmel，和老爷子简单聊了几句，表达了我的仰慕之情以及合影留念。老爷子说他去年开始已经不招PhD了，难怪我去年没申到。然而申到了我估计也不会去。老爷子问了我老板是谁，说他很好，这让我感到有点意外，因为老板并没有和Demmel合作过。

20号我的杂事比较多。早上仍然主要是在SCC19会场，中间偶尔也溜出去看看展览。去年第一次开会，刘姥姥进大观园，在展览区域逛个不停，都不知道是去开学术会议还是去锻炼健身。今年新鲜感大不如以前，也就不费那么多精力了。展览里有那个wafer size NPU的公司展出那块惊为天人的wafer size NPU，一块晶圆就是一块芯片，比脸还大，功耗15 kW。然而我觉得这个芯片并不太实用，因为单芯片性能太强，ML library很难优化支持，而且读数据进来处理的时候也会有瓶颈。



Rainmaker's Notebook

『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

Home

Archives

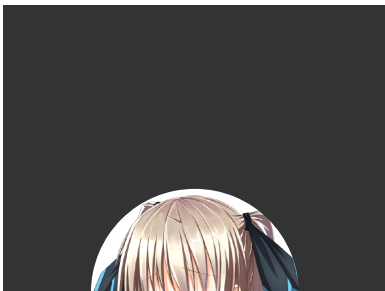
About

SiteXC



会场里另外一个让我觉得有点意思的是一个东京大学的研究组做的 NPU，只能做 CNN。但这芯片不仅支持单精度和半精度，还能跑双精度，而且单芯片有 35T 的双精度性能。这就非常有意思了：谁要用双精度来做神经网络呢？如果他们做 CNN 的方式是 im2col 然后走 GEMM 这一套，那么倒是有可能把这个芯片拿去跑 HPL，除此以外我真是再也想不到有什么用武之地了。下午1点半去听了另一篇做 symm SpMV 的论文，非常失望：预处理代价太高，实用性不强。下午2点半开始又要回去 SCC 会场守着，给来参观的高中生讲解。期间我也和丽狼&清华的同学吐槽了一下。周三下午是比赛最后的时间段，他们都累了，丽狼甚至在队伍大屏幕上放起了拌爱的视频。清华的惯例是在队伍大屏幕上放一个 2D 小车爬坡的程序，今天也放了一会儿《舌尖上的中国》。这大屏幕原本是给各队放队伍信息和宣传的。

5点半比赛结束，队员们可以走了，组委会开始改 report。我因为不是 application specialist，只要改 advisor' s report——最轻松而且没有技术含量的一部分。我改完以后8点就赶紧走去吃饭了，其他人还得继续改。其中，任务最重的是 reproducibility application 的评卷。RA 今年选了一个地震波相关的应用，是 SC18 的论文。这里面涉及的 polynomial filtering eigen solver 是祖师爷的算法，我之前搬砖



Rainmaker's Notebook

『求雨巫师的神奇之处在于他总是躲着不见你，却总说刚下完的雨是拜他所赐。』——《天真的人类学家》

[Home](#)

[Archives](#)

[About](#)

[SiteXC](#)

的时候也接触过。原本我跟 Rebecca 说我想负责这道题，然而最后负责这道题的组委会成员没有联系我给我派活儿。这题论文的一作是石嘉，开会的几天我倒是和他聊了不少。石嘉这次是真的敬业，各队的人有什么问题在 google group 里提出来以后他都很快进行了解答。20 号晚上改卷子他也是改到最晚才走，差不多9点半。

21 号是我最轻松的一天。早上基本在各处听论文。今年太湖光上的工作明显少了，而且没什么新意，都是新瓶装旧酒。听说杨广文老师正在把清华的人撤回去并且招一批新的人去无锡中心填缺。我个人并不看好太湖光及其后续升级版，主要是架构奇特（如果不用奇葩这个词的话），软件栈难用，官方技术支持缺位。相比之前，听说 Matrix2K 系列的加速卡还好用一点，无非是个基于 ARM 的众核加速卡，大概是中国版的 A64FX。下午去颁奖典礼，今年 GB 奖只有两个 finalist，其中一个是 Torsten Hoefler 团队的论文。Hoefler 大神今年全面开花，SC19 接收的 80 多篇论文里，他有份的就有 8 篇，几乎每一个 track 他都有一篇，还要再加上 GB finalist 这一篇。廖秋承跟我说 Hoefler 近年来就一门心思在 GB 奖上，结果今年终于得偿所愿了。至于 SCC，今年又是清华第一，我倒是不觉得意外。让不少人感到意外的是清华的 IO-500 也是第一，然而他们只用了一块 PCIe SSD 就干挺了 ETH Zurich（对，就是 Torsten Hoefler 做指导老师的队）用 Intel Optane 搭的阵列。

比起达拉斯会展中心，丹佛会展中心周围就是市区，吃的东西还是多很多的。原本以为今年能吃几顿好的，结果并没有时间，最后做了一个星期的精神京都民：附近有一家 Kyoto Ramen，那家店每天都是一堆挂着参会牌的亚洲面孔去吃，我5天吃了5顿。周三晚上最后一顿在那里吃的时候他们准备下班了，后厨传来了中文对话的声音，实在是太出戏了。嘛，起码中国人做的日本拉面要比美国人做的左宗棠鸡和汉堡要好下口一点。最后一晚组委会招待 technical program 的人去一个航空博物馆里吃饭开 party，还好去得早，要不然到后来排队都排死人了。一千多号人在场馆里只有6个小摊位提供食物，真不知道组委会是怎么想的。比起食物，倒是那些飞机还值得看一下。

周四晚上到周五下雪了，看到雪景算是赚到了。雪中的 Union Station 还挺好看的，让我想起了年初去北海道的时候。唉，一个人看雪景，难道不是太寂寞了吗，还是回去吧。

发布于 2019-12-16

/ tags: { [HPC](#) } { [Travel](#) } { [Conference](#) }

Related [Issues](#) not found

Please contact @EnigmaHuang to initialize the comment