CUBLAS FOR COLOR MATRIX

STEFAN ROISER

MADGRAPH DEVELOPERS, 31 OCT 2022
COLOR MATRIX MULTIPLICATION FOR $N$ EVENTS AND $M$ AMPLITUDES/COLORS IN PRECISION $T$

$N$ events, $M$ amplitudes $\times$ matrix $M \times M$ colors $\times$ $N$ evts, $M$ ampl (conj) $= N$ matrix elements

“original” implementation ($N$ times)

```
T me = 0;
for ( int i = 0; i < ncolor; ++i ) {
    complex<T> temp;
    for ( int j = 0; j < ncolor; ++j ) {
        temp += cm[i][j] * amp[j];
    }
    me += real(temp) * real(amp[i]) + imag(temp) * imag(amp[i]);
}
```

“cublas”
- split amplitudes into real and imaginary part
- call cublas with the multiplications above for each
- sum up real and imaginary results
CUBLAS IMPLEMENTATION DETAILS

1st multiplication: **cublasTsymm**

multiply symmetric (color) matrix with matrix (vectors for N events)

2nd multiplication: **cublasDgemvBatched**

multiply N batched matrices (1xM) with N vectors (size M).

NB: vector reorganised as a T**

Repeat the above for real and imaginary part and sum up the two.
Use of Tensor Cores

- Cublas decides itself whether to use tensor cores for a multiplication or not

- NB: for smaller matrices, e.g. 24 x 24 tensor cores not used at all

Same Matrix (120 x 120) calculation in:
- double precision
- single precision

<table>
<thead>
<tr>
<th>Calculation in:</th>
<th>Tensor Cores, Used</th>
<th>Tensor Cores, Not Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>double precision</td>
<td>4.466 ms</td>
<td>4.590 ms</td>
</tr>
<tr>
<td>single precision</td>
<td>1.088 ms</td>
<td>2.417 ms</td>
</tr>
</tbody>
</table>
COMPARISON ORIGINAL TO CUBLAS IMPLEMENTATION: COLOR MATRIX 120 X 120

e.g. gg→ttggg

runtime factors
tensor cores NOT used
tensor cores used
COMPARISON ORIGINAL TO CUBLAS IMPLEMENTATION: COLOR MATRIX 24 X 24

e.g. gg→ttgg

cublas original single precision
cublas original double precision

color matrix (24x24), single precision
color matrix (24x24), double precision

1 8k 16k 2M

gridsize

runtime

factors

1 8k 16k 2M

gridsize

runtime

factors

1 8k 16k 2M

gridsize

runtime

factors
MORE IDEAS

- Run with even bigger matrices, e.g. 6 particles in final state → color matrix 720x720
- I expect to use tensor cores with more computational load
- Tensor cores are only used by cublas for matrix - matrix multiplications
- Re-organise the second (currently batched) multiplication of matrix - vector to matrix - matrix, will it pay off?