GPU based photon propagation for CORSIKA 8

Dominik Baack
5th ICPPA
Online
09. 10. 2020
Corsika 8 Framework

- Successor of the well known Corsika 7 simulation
- Core completely written in C++

- https://gitlab.ikp.kit.edu/AirShowerPhysics/corsika
Highly parallel CORSIKA processing

~30 \gamma / m

32km

64km
<table>
<thead>
<tr>
<th>Function</th>
<th>CPU Time: Total</th>
<th>CPU Time: Self</th>
<th>Instructions Retired: Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>aamain</td>
<td>2232.101s</td>
<td>0.020s</td>
<td>100.0%</td>
</tr>
<tr>
<td>__libc_start_main</td>
<td>2232.101s</td>
<td>0s</td>
<td>100.0%</td>
</tr>
<tr>
<td>main</td>
<td>2232.101s</td>
<td>0s</td>
<td>100.0%</td>
</tr>
<tr>
<td>_start</td>
<td>2232.101s</td>
<td>0s</td>
<td>100.0%</td>
</tr>
<tr>
<td>box3</td>
<td>2208.159s</td>
<td>0.020s</td>
<td>99.0%</td>
</tr>
<tr>
<td>cerenk</td>
<td>2078.871s</td>
<td>533.937s</td>
<td>93.8%</td>
</tr>
<tr>
<td>egs4</td>
<td>2026.466s</td>
<td>0s</td>
<td>90.8%</td>
</tr>
<tr>
<td>em</td>
<td>2026.466s</td>
<td>0s</td>
<td>90.8%</td>
</tr>
<tr>
<td>shower</td>
<td>2026.466s</td>
<td>0.070s</td>
<td>90.8%</td>
</tr>
<tr>
<td>electr</td>
<td>2023.865s</td>
<td>34.329s</td>
<td>90.7%</td>
</tr>
<tr>
<td>distip</td>
<td>449.052s</td>
<td>203.941s</td>
<td>16.8%</td>
</tr>
<tr>
<td>__ieee754_acos_sse2</td>
<td>358.305s</td>
<td>318.770s</td>
<td>21.0%</td>
</tr>
<tr>
<td>tofp</td>
<td>336.495s</td>
<td>214.020s</td>
<td>15.5%</td>
</tr>
<tr>
<td>rmmard</td>
<td>250.622s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>__cos_avx</td>
<td>192.107s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rhof</td>
<td>190.668s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>updatc</td>
<td>187.214s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>__Gl__exp</td>
<td>180.922s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>update</td>
<td>180.016s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>__ieee754_exp_avx</td>
<td>166.768s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mutrac</td>
<td>165.054s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>do_sincos_1</td>
<td>110.374s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>__sin_avx</td>
<td>104.359s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>telout_</td>
<td>55.212s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>do_sincos_1</td>
<td>47.284s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>__doasin</td>
<td>37.784s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>do_cos</td>
<td>32.920s</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
42 Years of Microprocessor Trend Data

Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/
https://creativecommons.org/licenses/by/4.0/

GPU based photon propagation for CORSIKA 8

Dominik Baack
dominik.baack@tu-dortmund.de
High parallel Cherenkov light

- Completely independent calculations for each photon
- “Simple” arithmetic no complex processes

- OpenCL allows platform independent vectorization & parallelization, first test shows good performance
  ... but ...
- CUDA – Nvidia proprietary but higher performance possible
Prerequisite

- Linear particle tracks
- \( \gg 1024 \) track segments accumulated to reach good performance
Geometry

CTA Observatory - https://www.cta-observatory.org/about/how-cta-works/optimized-cta_orm_comp_webupdated_1800x590/

Dominik Baack
dominik.baack@tu-dortmund.de

GPU based photon propagation for CORSIKA 8
GPU based photon propagation for CORSIKA 8

Track Handling
Filter 1
Photon generation
Filter 2
Propagation & Collision

CPU

GPU

Particle tracks

Photon impact points and angles

Dominik Baack
dominik.baack@tu-dortmund.de
Computing Structure

- All Tracks to GPU
  - More work to GPU
  - Less data transfer

- Load, convert and distribute tracks to warps

- Split tracks if necessary partial

\[ \beta_{\text{start}} < 1 < \beta_{\text{end}} \]
\[ \beta_{\text{end}} < 1 < \beta_{\text{start}} \]

- Remove any tracks with \( \beta < 1 \)
Computing Structure

- Remove particle \( \vec{v} \cdot \vec{p} \leq \text{cut} \)
- Angular cut not possible with
  - diffuse Emission
  - Fluorescence

GPU based photon propagation for CORSIK A 8
GPU based photon propagation for CORSIKA 8

Computing Structure

- Track Handling
  - Filter 1
  - Filter 2
  - Photon generation
  - Propagation & Collision

- Tables in texture memory

- Use presolved Frank-Tamm Formula

\[
\frac{d^2 N}{dx d\lambda} = \frac{2\pi z^2}{\lambda^2} \cdot \sin^2(\theta_c)
\]

to calculate N and \(\alpha\)
Computing Structure

- Track Handling
- Filter 1
- Photon generation
- Filter 2
- Propagation & Collision

GPU based photon propagation for CORSIKA 8
Computing Structure

Track Handling
  Filter 1
  Photon generation
  Filter 2
  Propagation & Collision

Table in texture memory

GPU

GPU based photon propagation for CORSIKA 8
Computing Structure

Track Handling
  Filter 1
  Photon generation
  Filter 2
  Propagation & Collision

GPU

GPU based photon p

Dominik Baack
dominik.baack@tu-dortmund.de
Computing Structure

- Track Handling
  - Filter 1
  - Photon generation
  - Filter 2
  - Propagation & Collision

GPU
Propagation linear in first order ...

... but corrections necessary for modern IACT's
Computing Structure

Track Handling
  Filter 1
  Photon generation
  Filter 2
  Propagation & Collision

GPU

GPU based photon propagation for CORSIKA 8
GPU based photon propagation for CORSIKA 8

Dominik Baack
dominik.baack@tu-dortmund.de
Computing Structure

- Track Handling
  - Filter 1
  - Photon generation
  - Filter 2
- Propagation & Collision
- Tables in texture memory

GPU based photon propagation for CORSIKA 8
Computing Structure

- Propagation linear in first order
- Tabulated correction factors generated offline and applied on GPU

GPU based photon propagation for CORSIKA 8
Conclusion & Outlook

- GPU based light propagation possible and likely fast than CPU based approach

- Corsika 8 is an open source and modular framework for cosmic ray simulation: https://gitlab.ikp.kit.edu/AirShowerPhysics/corsika

- In Vivo tests still necessary with very new EM-Model (Proposal) for Corsika8

- Improvement by sorting to reduce “starvation” of threads or more efficient thinning
- Dedicated test of a more efficient fluorescence propagation