

GPU-accelerated Calculations of Hadron Spectra from Heavy-Ion Collisions

Matthew Golden
Advisor: Dr. Ulrich Heinz

August 11, 2017

1 Introduction

Heavy-ion collisions pose a unique theoretical and computational challenge. For each event, two heavy nuclei collide at large energies and leave behind a fireball of nuclear matter. The fireball has a temperature over 145 MeV or 2 trillion Kelvin. At these temperatures, hadrons cannot form, instead the fireball is a quark-gluon plasma (QGP). Computationally, an initial fireball is evolved as a viscous fluid; it cools as it expands until it falls below the 145 MeV temperature threshold and determines a chemical freeze-out surface. Hydrodynamic variables are converted to hadronic spectra. Many unstable resonances are produced through this condensation. These resonances decay to stable hadrons like pions and protons before a final spectrum is produced.

To make computational progress, a large number of events needs to be simulated. The observables from the simulated collisions form a statistical ensemble to compare to experiment. It is very important to make computation time small so more events can be computed and better statistics can be generated. The iEBE package provides a complete resource for simulating a collision from generating initial conditions to calculating observables and a final particle spectrum. iSpectra is contained within iEBE and is used to convert chemical freeze-out surfaces to hadron spectra. We have used CUDA C to accelerate these calculations on a GPU from a boost invariant surface.

Throughout this paper, we use the convention that $g_{00} = 1$ and choose units such that $c = \hbar = 1$.

2 Relativistic Hydrodynamics

The stress-energy tensor $T^{\mu\nu}$ and the equation of state for the QGP contain all of information of the fluid. The evolution of $T^{\mu\nu}$ is not relevant to our work, but the meaning of its components is. In viscous hydrodynamics, the stress-energy tensor can be split using the timelike eigenvector $T^{\mu\nu}u_\nu = \epsilon u^\mu$, where

we define u^μ to be the fluid velocity and ϵ to be the energy density. We can then decompose the tensor with its eigenvector.

$$T^{\mu\nu} = \epsilon u^\mu u^\nu - (P + \Pi)\Delta^{\mu\nu} + \pi^{\mu\nu}. \quad (1)$$

P is the thermodynamic pressure and Π is the bulk viscous pressure. $\Delta^{\mu\nu} = g^{\mu\nu} - u^\mu u^\nu$ is a projection tensor orthogonal to the fluid velocity. $\pi^{\mu\nu}$ is the shear viscosity tensor. It is orthogonal to the fluid velocity $\pi^{\mu\nu}u_\nu = 0$ and is traceless $\pi_\mu{}^\mu = 0$.

These variables are evolved according to some hydrodynamic equations, and the QGP equation of state is used to find the temperature of the fluid. The QGP cools as it expands, and the conversion surface at 145 MeV is recorded.

3 Thermal Spectra

This 145 MeV surface marks the phase transition back to hadronic matter. This phase transition is a smooth cross-over, so the rigid cutoff is computationally imposed rather than physical. The challenge of converting hydrodynamic variables into hadron spectra was solved with the Cooper-Frye formula [1].

$$E \frac{dN}{d^3p} = \frac{g}{(2\pi)^3} \int_\Sigma d^3\sigma_\mu(x) p^\mu f(x, p). \quad (2)$$

The momentum distribution $\frac{dN}{d^3p}$ known from nonrelativistic physics is multiplied by energy to make it a Lorentz scalar, since $d^4p \delta(E^2 - p^2 - m^2)\Theta(E) = \frac{d^3p}{2E}$ is manifestly Lorentz invariant. The integral (2) is over the entire conversion surface Σ (with surface normal vector $d^3\sigma_\mu(x)$ at point x) and depends on the distribution function $f(x, p)$. The distribution function is split into equilibrium and nonequilibrium parts, $f(x, p) = f_0(x, p) + \delta f(x, p)$, where the equilibrium distribution function is

$$f_0(x, p) = \frac{1}{e^{\beta u_\mu p^\mu} \pm 1}. \quad (3)$$

$u_\mu p^\mu$ is the energy of the particle in the rest frame of the fluid, and $\beta = (kT)^{-1}$. The sign in the denominator is determined by the statistics of the hadron: -1 for a boson and $+1$ for a fermion. In the fluid rest frame where $u^\mu = (1, 0, 0, 0)$, these reduce to the standard Bose-Einstein and Fermi-Dirac distributions, but we have expressed them covariantly so that they hold in any reference frame where the fluid cell moves with four-velocity u^μ .

The nonequilibrium deviation $\delta f(x, p)$ arises from the viscous correction to the stress energy tensor. It is some function of $\pi^{\mu\nu}$ and Π determined by which approximation is being used for viscous hydrodynamics and to what order these corrections are considered.

4 Resonance Decays

After a hadron is emitted from the fireball, it will not necessarily live long enough to be detected. Heavy resonances will generally decay to lighter stable hadrons like pions or protons.

We consider a general n -body decay in the rest frame of the parent resonance. The momentum of the parent is P^μ and momentum of the considered daughter particle is p^μ . The other $n - 1$ daughter particles have momentum p_i^μ , $i = 2, \dots, n$. The decays are required to conserve energy-momentum and obey the mass relations $p_i^\mu p_{i\mu} = m_i^2$. Spin should also be conserved, but we take the spinless approximation.

The energy and momentum of the the daughter particle of interest can be written in terms of the invariant quantity $s = (p_2^\mu + \dots + p_n^\mu)^2$. In the parent rest frame, we write the daughter's energy and momentum magnitude E^* and p^* .

$$E^* = \frac{M^2 + m^2 - s}{2M}, \quad (4)$$

$$p^* = \sqrt{\frac{[(M + m)^2 - s][(M - m)^2 - s]}{4M^2}}. \quad (5)$$

We make the approximation that decays are purely kinematic, and determined completely by conservation laws. The decay phase space is a distribution of s written $g(s)$. The phase space is normalized to be the branching ratio b of the decay. From this, we can derive the contribution to the daughter spectrum [2].

$$\frac{dN}{p_T dp_T d\phi dy} = N \int ds g(s) \int \frac{d^3 P}{E_r} \delta(p_\mu P^\mu - ME^*) \frac{dN_r}{P_T dP_T d\Phi dY}. \quad (6)$$

Here we have defined the left hand side to be the contribution to invariant spectra. $p_T^2 = p_x^2 + p_y^2$, $\phi = \arctan \frac{p_y}{p_x}$, and the rapidity $y = \operatorname{arctanh} \frac{p_z}{E}$. The parent resonance has corresponding capital variables P_T , Φ , and Y . It can be shown the left hand side is equivalent to $E \frac{dN}{d^3 p}$. N is a normalization constant for $g(s)$. The delta function in the momentum integral ensures the correct energy in the resonance rest frame.

In figure 1, the thermal and total spectra are plotted against p_T for different stable hadrons. For high p_T , the spectra fall off exponentially. The proton spectra overtakes the mesons at high enough p_T due to the effect of radial flow which pushes the heavier protons more strongly than the lighter pions to higher transverse momenta.

These spectra can be integrated to obtain angular probability distributions $P(\phi)$. Simulating collisions with fluctuating initial conditions gives a varying set of these $P(\phi)$. The probability distributions themselves should not be compared to experiment directly, rather one should use the complex Fourier expansion coefficients v_n [3].

$$P(\phi) = \frac{1}{2\pi} \left[1 + \sum_{n=1}^{\infty} (v_n e^{-in\phi} + v_n^* e^{in\phi}) \right]. \quad (7)$$

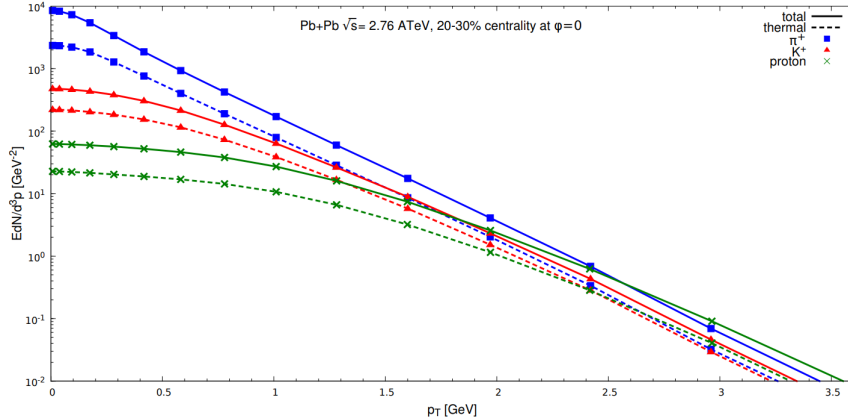


Figure 1: Invariant spectra of pions, kaons, and protons as a function of transverse momentum p_T , emitted with azimuthal angle $\phi = 0$ relative to the direction of the impact parameter.

5 GPU Acceleration

A GPU provides immense computational power for parallel problems. A single GPU contains thousands of cores capable of numerical manipulation. They do not work completely independently, rather groups of cores called warps carry out the same instructions in parallel. This allows parallelizable tasks to be accelerated significantly. A parallelizable task is one with many computations that can be done independently of one another.

The calculation of hadron spectra from a conversion surface is such a problem. There are many ways one could parallelize this problem. The straightforward way is to parallelize over momentum space. The hadron spectra at different points of momentum space are independently computed with the Cooper-Frye formula. This is not a good way to parallelize because there is a large amount of information that would need to be referenced during loops over the surface. Instead, it is better to parallelize over the surface. Since there are 35,000 surface cells and 500 points in momentum space for a particle, it made more sense to parallelize over this larger set. The contributions to the spectra from each cell are calculated in parallel by associating a cell with each thread on the GPU. They are then added using parallel reduction, which scales like $\log(n)$ instead of n . This reduction is carried out in two steps: a summation over all threads in a block (a collection of communicating threads) and then a summation over blocks.

An additional modification is the abandonment of an approximation used in the original code. For numerical efficiency, the old CPU code grouped similar particles with close mass and chemical potential and computed a single spectra for these groups. This decreased computation time on the CPU by a factor of 5, but it is an approximation that ultimately lead to a systematic error of

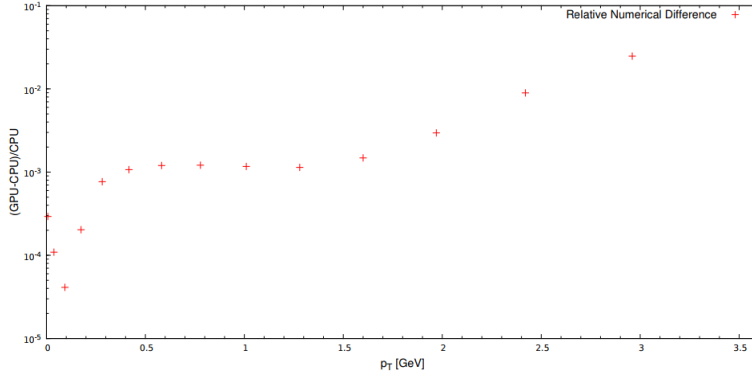


Figure 2: $|\frac{GPU-CPU}{CPU}|$ for the pion spectra at $\varphi \approx 0$. The differences are on the order of numerical precision.

several percent. Since on the GPU the computation time has been lowered, it is no longer necessary to make this approximation, thus eliminating this small systematic error.

The computation of decays of different resonances is, however, not independent. If a resonance decays to a secondary resonance before finally decaying to a stable particle, the first decay process should be computed before the second. The parallelization should instead be carried out for independent decays. For example, all direct decays to a proton are independent, and should be done in parallel. There is a secondary parallelization that is also applied by parallelizing over momentum space. This still has impact on the computation time of these decays, but the nature of the problem requires some time for monitoring when certain computations are complete.

Previously, the calculation of thermal spectra and subsequent decays took 31 minutes and 4 minutes, respectively, for a surface with 35,000 cells. This was for unparallelized code running on a single CPU core. After parallelizing over the cells of the surface, the thermal spectra was calculated in 16.5 seconds. The resonance contributions could only be parallelized to a certain extent, and went down to 20 seconds. Overall, the process was reduced from 35 minutes to 36 seconds. This is an acceleration factor of 60, and will increase with surface size. As surface size increases, the acceleration of thermal calculations will contribute more to the overall acceleration and asymptotically approach an acceleration factor around 100

This code was cross checked with CPU results, and fell within an acceptable deviation. In figure 2, we plot the normalized difference $|\frac{a-b}{b}|$ where a is the GPU result and b is the CPU result. This difference is of the order of one per mille over most of the p_T range but increases to about 10% for p_T around 3.5 GeV. The source of this error is not definitely known, but it is suspected that contribution may arise from the way sums are carried out on the CPU. On a single CPU core, the sums are serial, and a small number is added to a much

larger running sum. However, on a GPU, the sums are parallel reductions which add numbers of the same order of magnitude and reduce the impact of floating point errors. In this sense, the GPU produces more reliable sums, but it is unknown if this is the sole contributor to numerical differences.

6 Conclusion

GPU computation is powerful for parallel problems. Computing these spectra faster allows for a GPU to compute over two thousand spectra in a day if it is provided the surfaces to work with. This will help produce larger theoretical data sets for comparison with the (much larger) sets of experimental data, allowing for a more precise analysis of heavy-ion collisions. This contributes to the overall goal of the JETSCAPE collaboration to create a user-friendly and efficient code package for simulating heavy-ion collisions.

References

- [1] F. Cooper, G. Frye: Phys. Rev. D 10, 186, 1974.
- [2] J. Sollfrank, P. Koch, U. Heinz: Phys. Lett. B252, 256, 1990.
- [3] Sooraj Radhakrishnan *Measurement of the correlation between flow harmonics of different order in leadlead collisions at $s_{NN}=2.76$ TeV with ATLAS*. Nuclear Physics A, 956, 328-331, 2016.