# Introduction to the Full Bandwidth Amplitude Analysis Software——*FALLS*

张然[1]，蔡浩[1]，朱凯[2]

hcai@whu.edu.cn

[1]武汉大学
[2]中科院高能物理研究所

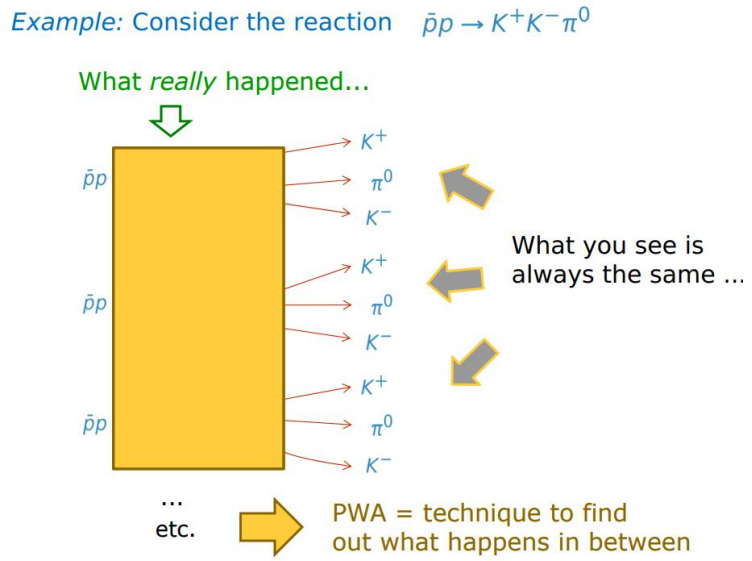
# 提纲

1. 背景介绍
2. 满带宽振幅分析软件——*FALLS*
   (FULL BANDWIDTH AMPLITUDE ANALYSIS SOFTWARE)
3. 其他问题
4. 总结

# 1. 背景介绍

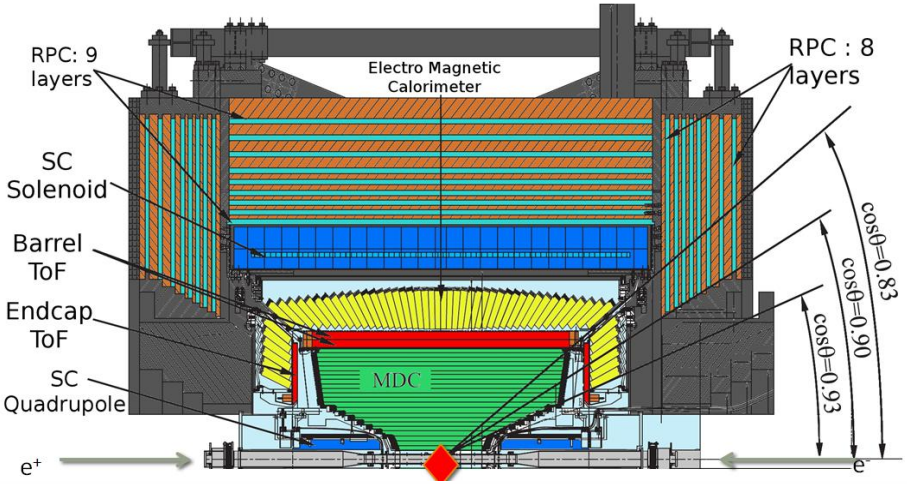
# 分波分析 (Partial Wave Analysis)

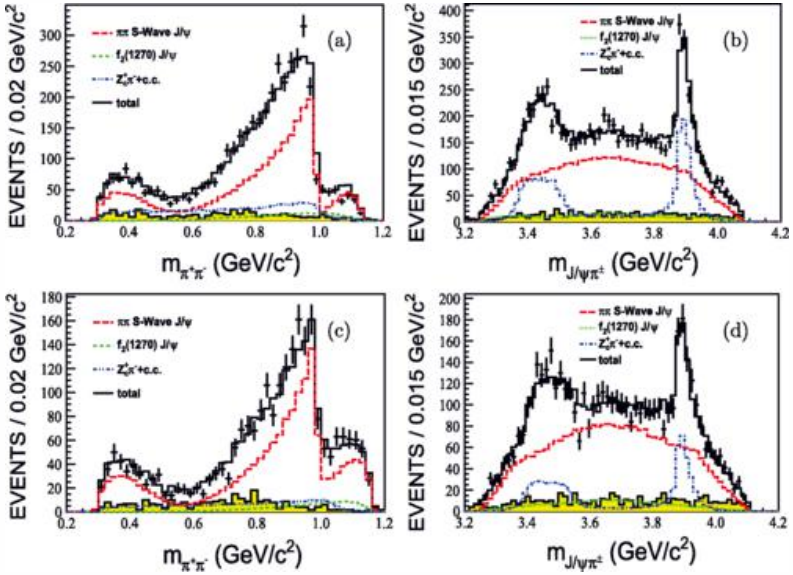
- 分波分析（Partial Wave Analysis）利用事例全部物理信息，直接拟合振幅，可以处理共振态的干涉、叠加，可以精确测量强子共振态的自旋-宇称、质量、宽度、分支比，是研究强子谱学的重要工具。
- 分波分析使用的数据数据量大，计算复杂，十分费时。



# 北京谱仪III实验 (BESIII)

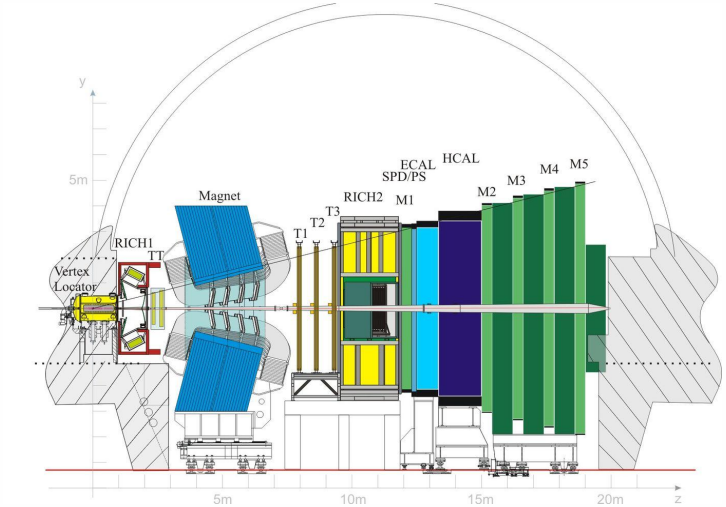
- “BESIII发现Zc(3900)”入选2013年度“中国科学十大进展”

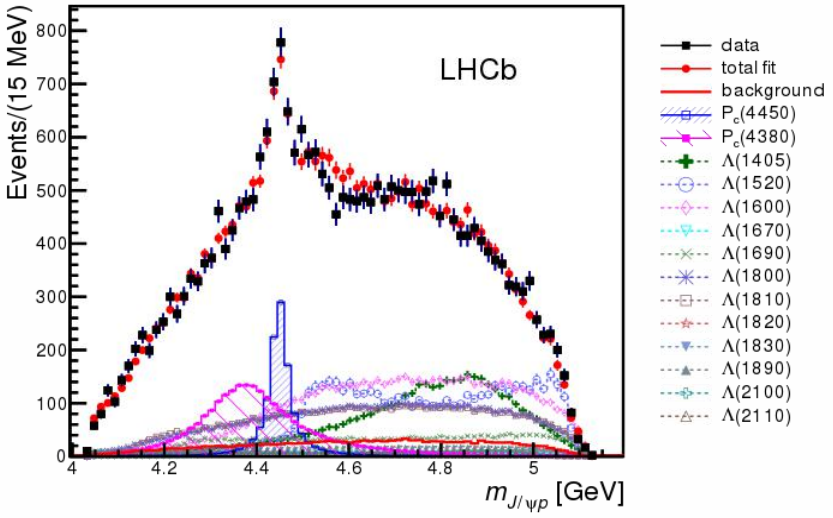


# 大型强子对撞机底夸克实验 (LHCb)

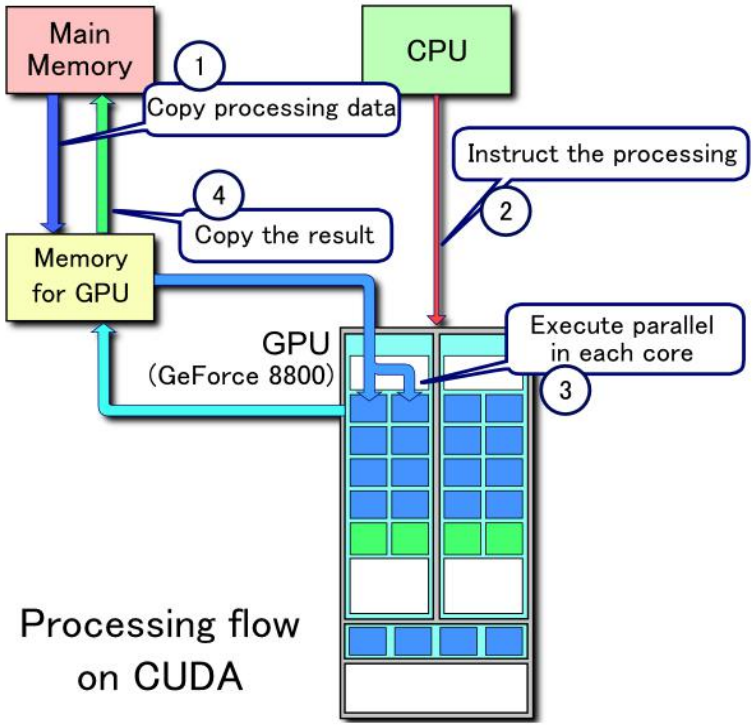
- LHCb发现五夸克态入选2015年英国物理学会公布的年度国际物理学领域的十项重大突破（Breakthrough of the Year）



# CUDA

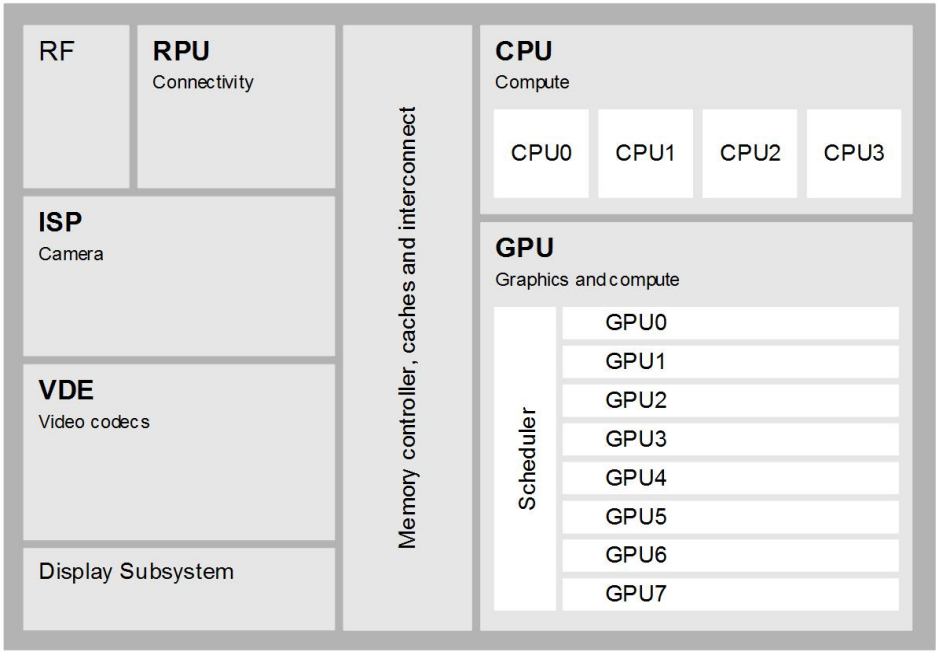
- CUDA is a parallel computing platform and application programming interface (API) model created by Nvidia.
- The CUDA platform is designed to work with programming languages such as C, C++, and Fortran.
- CUDA SDK support for different compute capability (Kepler, Maxwell, Pascal, Volta, Turing...)



# 异构计算 (Heterogeneous computing)

- 将计算复杂，数据量极大，易于并行的似然值计算部分交由GPU计算
- 将串行执行的控制流以及少量的数据操作交由CPU进行

算法结构的优化非常重要！

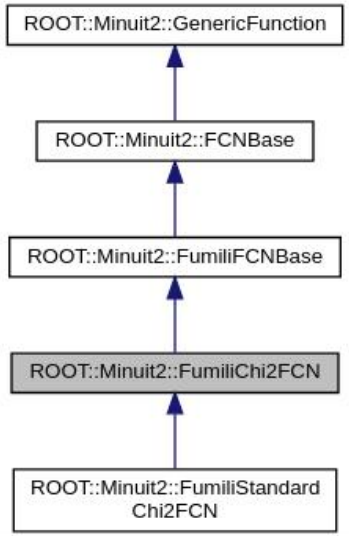


# NVidia Tesla V100


| | Tesla V100 PCIe | Tesla V100 SXM2 |
|---|---|---|
| GPU Architecture | NVIDIA Volta | |
| NVIDIA Tensor Cores | 640 | |
| NVIDIA CUDA® Cores | 5,120 | |
| Double-Precision Performance | 7 TFLOPS | 7.5 TFLOPS |
| Single-Precision Performance | 14 TFLOPS | 15 TFLOPS |
| Tensor Performance | 112 TFLOPS | 120 TFLOPS |
| GPU Memory | 16 GB HBM2 | |
| Memory Bandwidth | 900 GB/sec | |
| ECC | Yes | |
| Interconnect Bandwidth* | 32 GB/sec | 300 GB/sec |
| System Interface | PCIe Gen3 | NVIDIA NVLink |
| Form Factor | PCIe Full Height/Length | SXM2 |
| Max Power Comsumption | 250 W | 300 W |
| Thermal Solution | Passive | |
| Compute APIs | CUDA, DirectCompute, OpenCL™, OpenACC | |

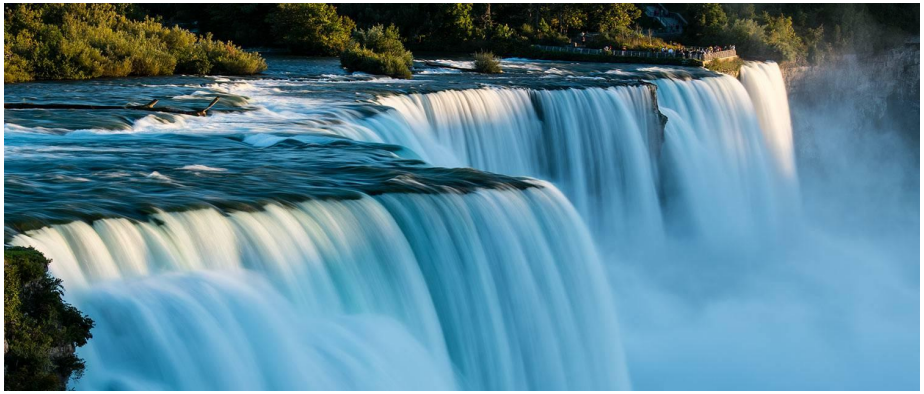
# ROOT and MINUIT2

- ROOT provides all the functionalities needed to deal with big data processing, statistical analysis, visualisation and storage.
- It is mainly written in C++ but integrated with other languages such as Python and R.
- MINUIT, now MINUIT2, is a numerical minimization computer program originally written in the FORTRAN programming language by CERN staff physicist Fred James in the 1970s.
- The program searches for minima in a user-defined function with respect to one or more parameters using several different methods as specified by the user.
- Quasi-Newton methods are mainly methods used to either find zeroes or local maxima and minima of functions in MINUIT.
- The new MINUIT is an optional package (minuit2) in the ROOT release. As of October 2014 the latest version is 5.34.14, released on 24 January 2014.

3. 满带宽振幅分析软件——*FALLS*
(FULL BANDWIDTH AMPLITUDE ANALYSIS SOFTWARE)

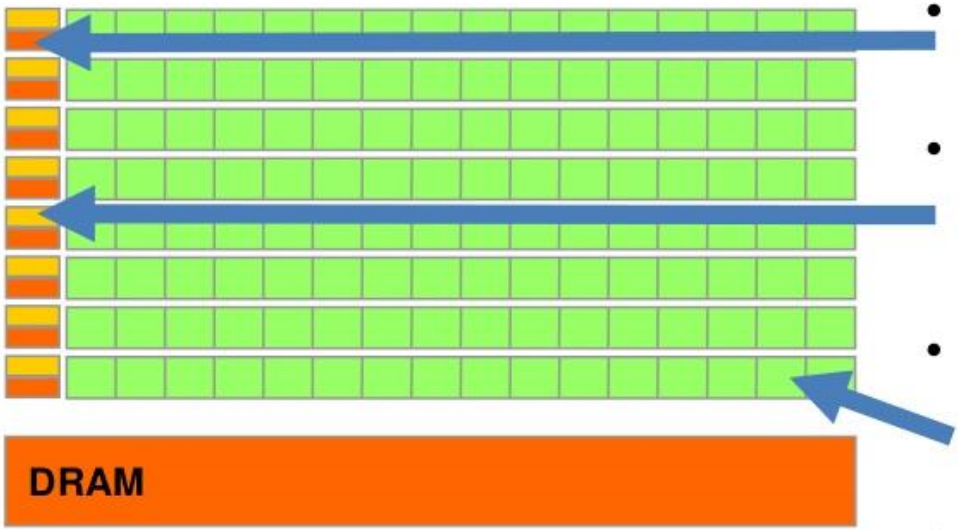
# FALLS 的目标



# 振幅分析软件的主要计算瓶颈

- 要处理的事例数非常巨大
- 拟合模型非常复杂，参数空间巨大
  - 需要用大量蒙特卡洛积分来保证概率函数的归一化
- 拟合算法需要的计算步数非常多

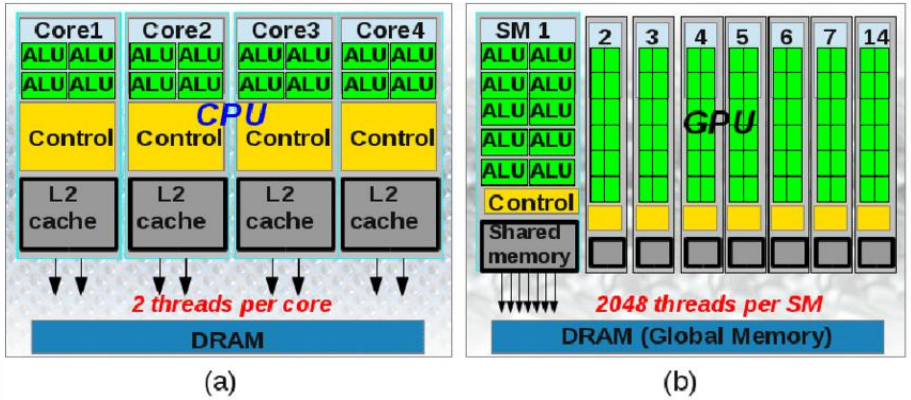
# GPU计算架构特点




- Small caches
  - To boost memory throughput
- Simple control
  - No branch prediction
  - No data forwarding
- Energy efficient ALUs
  - Many, long latency but heavily pipelined for high throughput
- Require massive number of threads to tolerate latencies

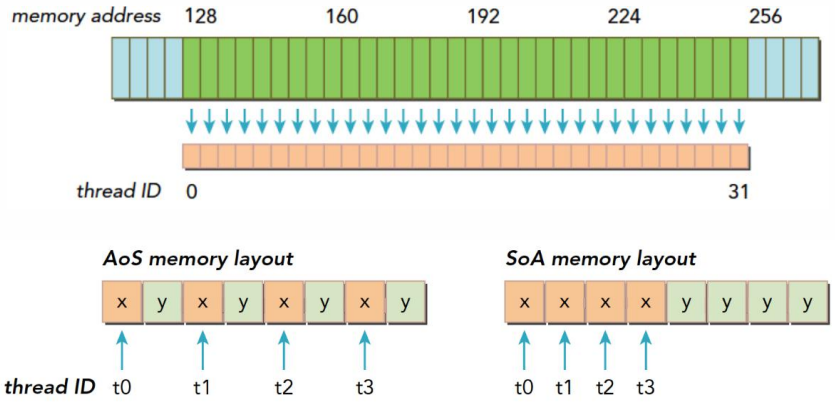
# 程序优化：充分调动计算单元

- 拆掉大的数据结构。
- 将计算部分的CUDA代码全部分解成小的片段，提高计算效率。
- 将条件分支全部移到程序外部，交给CPU处理。

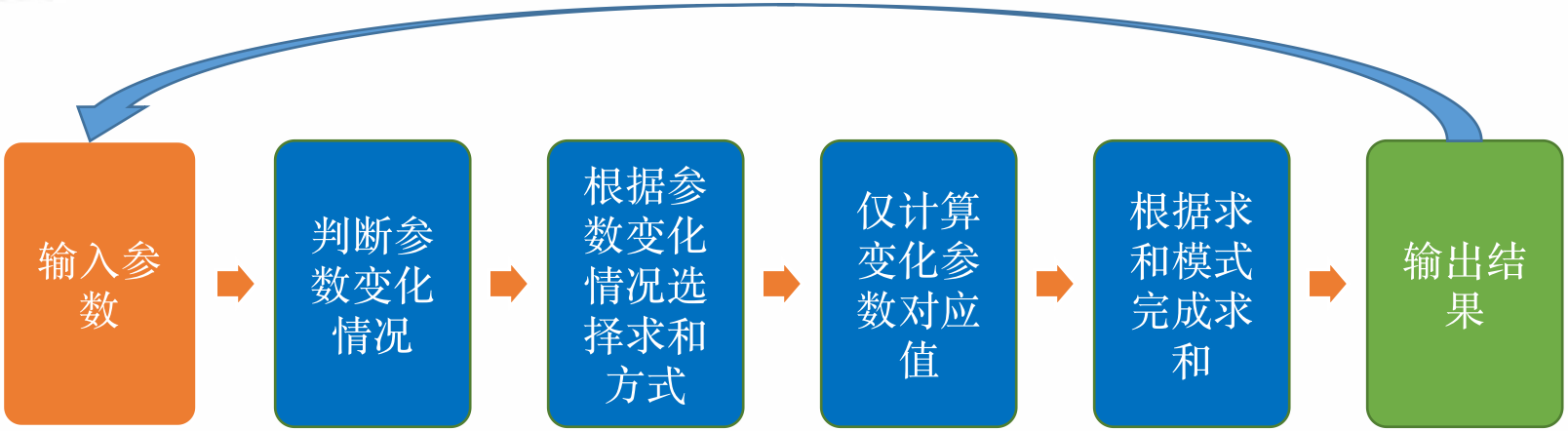


# 程序优化：数据对齐

- 当一个内存事务的首个访问地址是缓存粒度（32或128字节）的偶数倍的时候称为对齐内存访问，非对齐的内存访问会造成带宽浪费。
- 当一个线程束（warp）内的线程访问的内存都在一个内存块里的时候，就会出现合并访问
  - 一次内存操作完成所有线程（thread）的读取请求。
- 在涉及到复数计算时，如果简单地将数据以复数形式（连续两个double变量）存储，会浪费一半内存带宽。



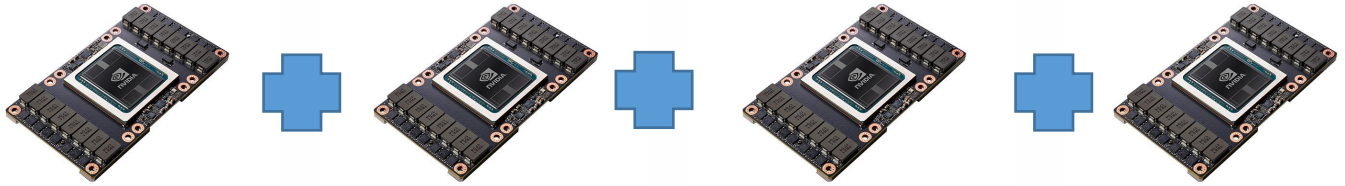
# 程序优化：算法优化




- 由于计算程序的细粒度的拆分，可以精确控制计算需求，算法优化效果显著。
- 所有的求和计算都在GPU中进行，GPU和CPU之间的数据交互降到最低。

# 多GPU联合计算

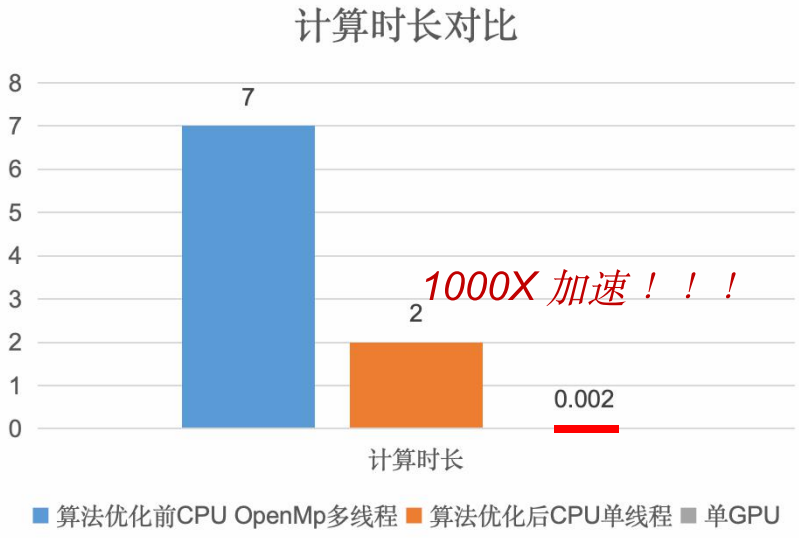
- *FALLS*已经实现了多GPU联合计算，并且实现了线性加速。



1 + 1 + 1 + 1 = 4

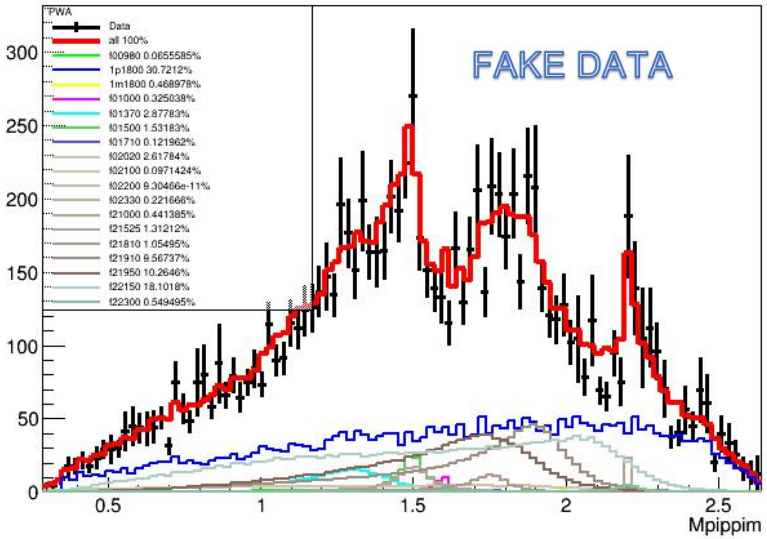
# 计算速度测试

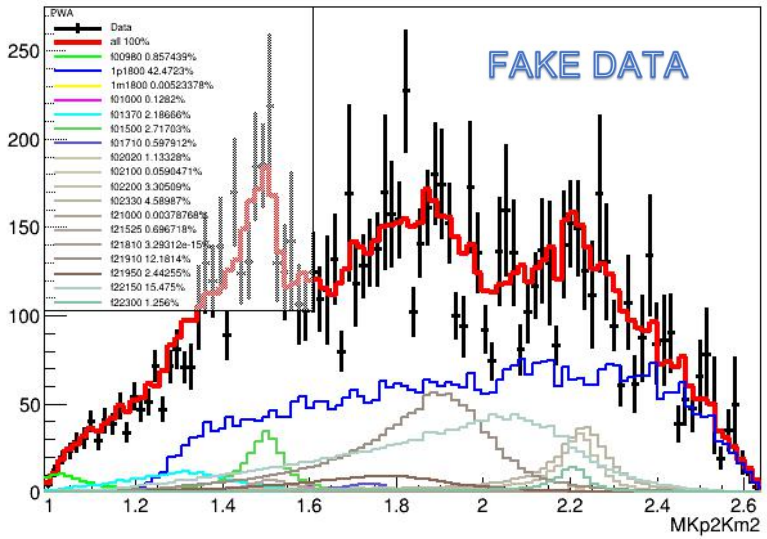
- *FALLS* 可以处理更大的数据量和更大的参数空间
- 例如处理1.6万事例，并用40万蒙特卡洛事例对参数空间进行积分，参数空间维度为190，使用单个Tesla V100，迭代步数50万步，程序总时长为50分钟。



# 拟合结果测试

- *FALLS* 可以进行关联拟合，并且已经通过了输入输出检查。

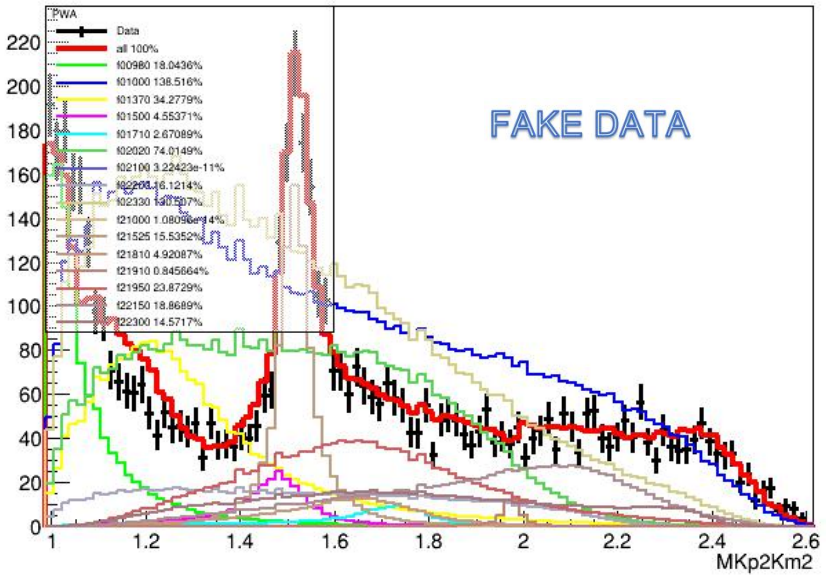


# 4. 其他问题

# 过拟合 (overfitting)

- 开启更大的参数空间后，过拟合现象变得比较突出。
- 一些不应该存在的中间态通过复杂的干涉效应相互抵消存在于最后的结果中。

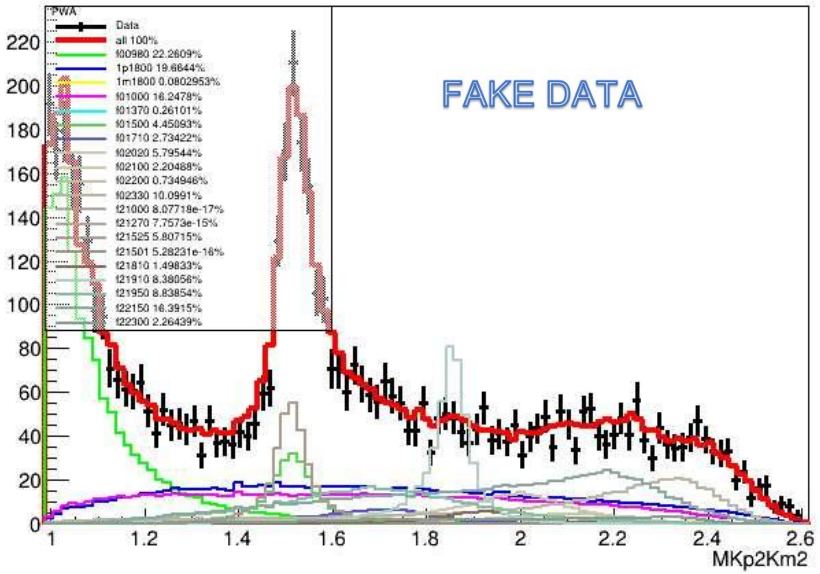


# LASSO算法

- Lasso算法（least absolute shrinkage and selection operator）是一种同时进行特征选择和正则化（数学）的回归分析方法，旨在增强统计模型的预测准确性和可解释性。

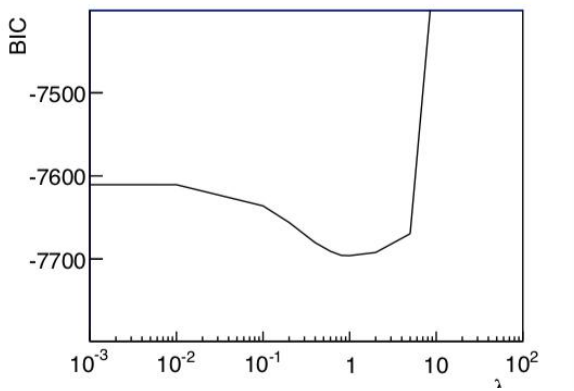
  *Regression shrinkage and selection via the Lasso, Robert Tibshirani, J. R. Statist. Soc. B(1996) 58, No. 1, pp.267-288*

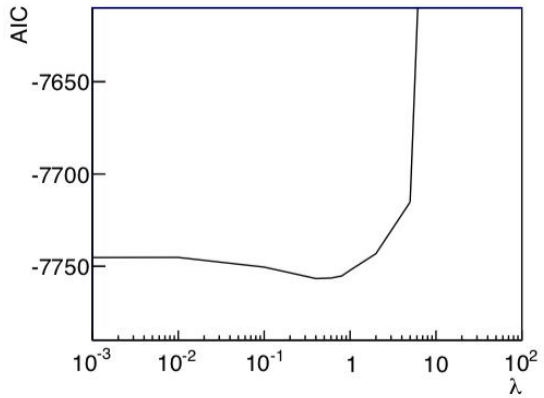
- 通过引入LASSO惩罚项，我们可以压缩无关参数的大小，选择出我们需要的中间态，抑制过拟合现象。



# 信息熵扫描

- 引入LASSO算法后，惩罚项系数极大影响了参数选择的效果。
- 为了选择出最好的结果，我们需要找到最优的惩罚项系数。
- 要多次调整系数后进行拟合，根据信息熵找到最优的惩罚项系数。



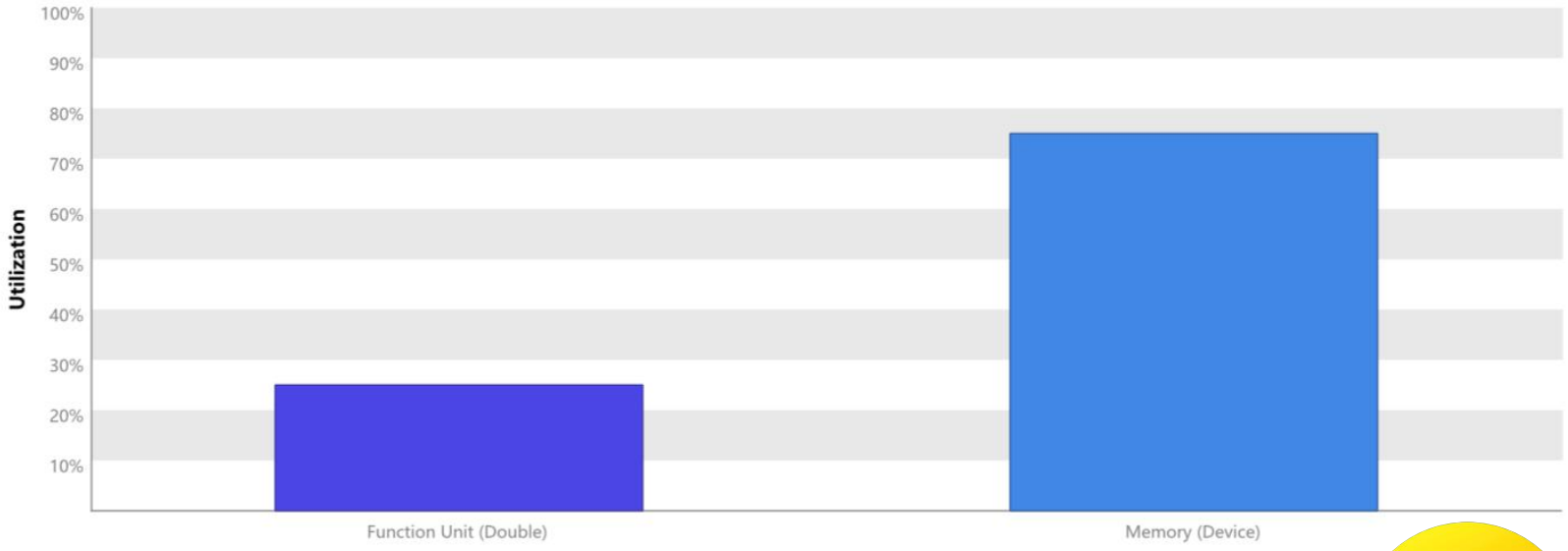

$$\mathrm{AIC}(\lambda) = -2\log\mathscr{L} + 2r, \qquad \mathrm{BIC}(\lambda) = -2\log\mathscr{L} + r\log n,$$

Model selection for amplitude analysis B. Guegan, J. Hardin, J. Stevens and M. Williams, arXiv:1505.05133

对算力需求极大!

# FALLS 真的满带宽吗?

- 利用 nvprof 软件可以得到 NVIDIA GPU 完整运行报告。
- Tesla V100-SXM2-16GB 的理论内存带宽是 900GB/s




我们似乎只占用了总带宽的 75%

# Tesla V100的“最佳”带宽

Table 3.1: Geometry, properties and latency of the memory hierarchy on the Volta, Pascal, Maxwell and Kepler architectures. All data in this table are measured on PCI-E cards.

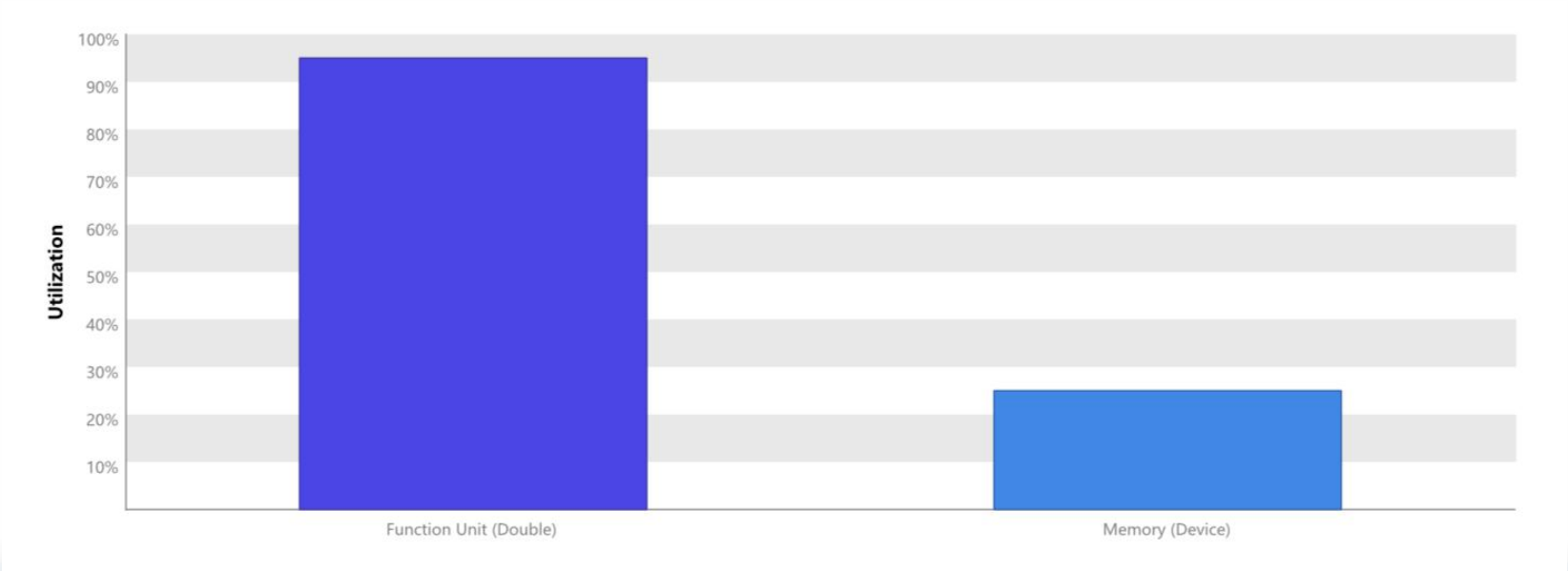
| | | Volta V100 GV100 | Pascal P100 GP100 | Pascal P4 GP104 | Maxwell M60 GM204 | Kepler K80 GK210 |
|---|---|---|---|---|---|---|
| Global memory | Memory bus | HBM2 | HBM2 | GDDR5 | GDDR5 | GDDR5 |
| | Size | 16,152 MiB | 16,276 MiB | 8,115 MiB | 8,155 MiB | 12,237 MiB |
| | Max clock rate ($f_m$) | 877 MHz | 715 MHz | 3,003 MHz | 2,505 MHz | 2,505 MHz |
| | Theoretical bandwidth | 900 GiB/s | 732 GiB/s | 192 GiB/s | 160 GiB/s | 240 GiB/s |
| | Measured bandwidth | 750 GiB/s | 510 GiB/s | 162 GiB/s | 127 GiB/s | 191 GiB/s |
| | Measured/Theoretical Ratio | 83.3% | 69.6% | 84.4% | 79.3% | 77.5% |

*Dissecting the NVIDIA Volta GPU Architecture via Microbenchmarking, Zhe Jia, Marco Maggioni, Benjamin Staiger, Daniele P. Scarpazza, arXiv:1804.06826*

*FALLS* 在 Tesla V100 上的确无愧于“满带宽”！

# *FALLS* 在游戏显卡上的表现

- 基于 GeForce GTX 1070 With Max-Q Design 进行测试，其理论带宽 250GB/s
- 由于双精度计算能力的限制，*FALLS* 在游戏显卡上对带宽的占用率普遍低于30%

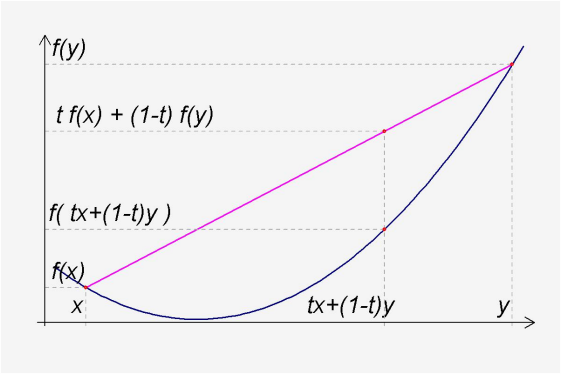


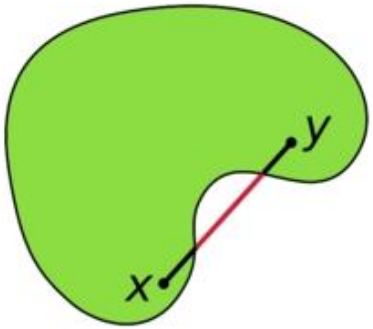
# *FALLS* 中CPU的运行效率

- CPU部分效率偏低
  - MINUIT对多CPU支持不佳
- 得益于CPU和GPU功能的良好分割，一块 Tesla V100 可以同时满速运行三个 *FALLS* 作业！

# 凸优化?

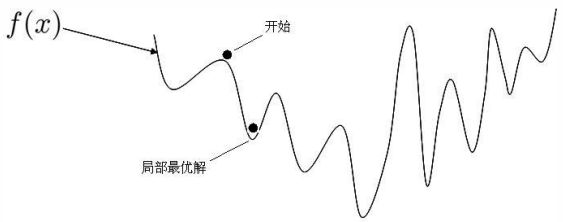
- 凸函数和非凸函数
- 非凸优化问题

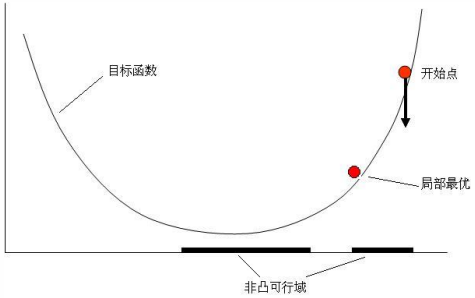



实际建模中判断一个最优化问题是不是凸优化问题一般看以下几点:

- 目标函数f如果不是凸函数，则不是凸优化问题
- 决策变量x中包含离散变量（0-1变量或整数变量），则不是凸优化问题
- 约束条件写成个g(x)<0时，g如果不是凸函数，则不是凸优化问题



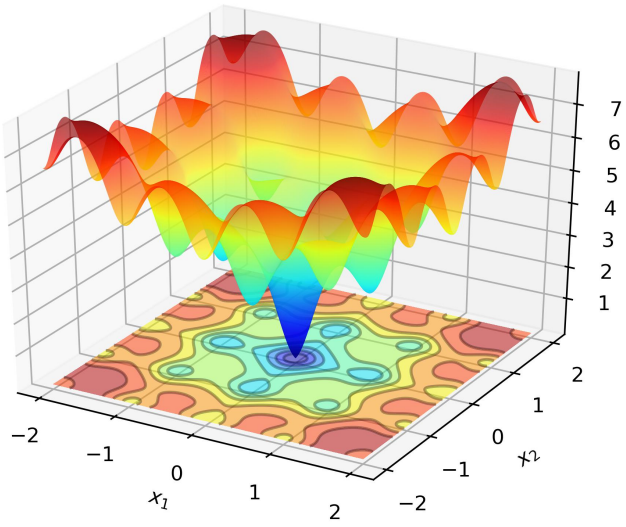



# 优化方法

- Quasi Newton *Upgraded in future version of RooFit*
  - 容易收敛到局部最优，容易被困在鞍点
- SGD（mini-batch gradient descent）
- Momentum
- Nesterov
- Adagrad
- Adadelta
- RMSprop
- Adam
  - 适用于大多非凸优化，适用于大数据集和高维空间 *Included in future version of TMVA*
- Adamax
- Nadam
- … …

优化软件包

- OptimLib
- nlopt
- … …



# 武汉大学超算中心简介

- CPU核心数>8000
- Intel KNL节点168台
- NVidia V100 408块（500块）
  - 每个node配置4块V100，提供NVLink连接
  - 每个node配置96G内存
- 硬盘存储 6PB

# 总结

- *FALLS* 对GPU内存带宽的利用率接近极限。
- *FALLS* 基本突破的振幅分析软件的计算能力瓶颈，特别是多GPU联合计算的实现，使计算速度不再成为制约振幅分析的主要困难。
- *FALLS* 有助于我们更深刻的研究过拟合、蒙卡积分对拟合的影响、非凸优化等对算力需求极大的复杂问题。
- *FALLS* 正在实际用于BESIII物理的振幅分析。
- 感谢武汉大学超算中心对本项目的大力支持！

谢谢大家！