

# 环境配置

---

## 1. 必备配置

---

- 符合c++ -17标准的现代c++编译器(gcc >= 7, Intel >= 19.0.1, clang >= 7.0)
- CMake (>= 3.13)
- Boost library (>= 1.58.0, 我们建议构建/安装完整的库)
- libllvm (>= 7.0 , 安装libllvm时需要开启RTTI, 但是如果使用直接编译好的, 则并不会对RTTI提供支持。因此选择下载源代码自己编译安装, 并设置 `-DLLVM_ENABLE_RTTI=ON` )
- libfmt (>= 4.0)

## 2. 可选配置

---

- Linux HUGE\_PAGES支持: (请参见[www.kernel.org/doc/Documentation/vm/hugetlbpage.txt](http://www.kernel.org/doc/Documentation/vm/hugetlbpage.txt))。如果不启用HUGE\_PAGES, 性能很可能会降低。DepGraph使用2MB的巨大页面大小, 并依赖于内核配置来留出大量的2MB页面。
- libnuma支持: 如果没有它, 性能可能会下降。请在Debian类系统上安装libnuma-dev, 在Red Hat类系统上安装numactl-dev。
- Doxygen (>= 1.8.5): 用于将文档编译为网页或latex文件。
- PAPI (>= 5.2.0.0 ): 用于分析代码。
- Vtune (>= 2017 ): 用于分析代码。
- MPICH2 (>= 3.2): 构建和运行分布式系统应用程序。
- CUDA (>= 8.0 and < 11.0) : 构建GPU或分布式异构应用程序。
- Eigen (3.3.1 works for us) : 一些矩阵补全应用变体。

# 文件说明

---

- build: 项目编译目录, 存放可执行文件
- libgalois: 包含共享内存DepGraph库的源代码 e.g., runtime, graphs, worklists, etc.
- lonestar: 包含Lonestar基准测试应用程序和DepGraph教程示例
- libdist: 包含分布式内存和异构DepGraph库的源代码
- lonestardist: 包含分布式内存和异构基准测试应用程序的源代码。请参考“lonestardist/README”。获取构建和运行这些应用程序的说明。
  - analytics
    - distributed
      - bfs: 分布式的宽度优先遍历算法

- tools: 包含各种辅助程序, 例如用于在图形文件格式之间进行转换的图形转换器和用于打印图形属性的图形统计
- libcusp: 包含了名为cusp的分区策略
- libgluon: 分布式通信库, 传递同步信息, 根据通信量的不同选择不同的通信策略
- libpangolin: 这是用于高效灵活的图挖掘的pangolin框架

## 编译命令

---

```
cd DepGraph运行目录 #进入项目的最顶层目录
mkdir build #如果没有build目录, 构建目录
cd build #进入build目录
cmake .. #生成makefile
cd lonestar\analytics\distributed\bfs #转移到bfs子目录
make -j #加速make编译
```

## 执行命令

---

### 1. 生成数据集

---

```
/home/ubuntu/lhy/run/Myproject/generator_graph/src/generator_one_graph 10 10
/home/ubuntu/lhy/run/Myproject/dataset/dataset3.bin
```

#### 1.1 参数解释

generator\_one\_graph: 生成数据集的可执行文件

10: 顶点数 $n$  ( $2^n$ )

10: 边数 $m(n*m)$

dataset3.bin: 生成数据集位置

### 2. 数据集格式转化

---

```
/home/ubuntu/lhy/run/build/tools/dist-graph-convert/dist-graph-convert --
edgeType=void --bin2gr --tempDir=/home/ubuntu/lhy/run/Myproject/temp -
numNodes=1024 /home/ubuntu/lhy/run/Myproject/dataset/dataset3.bin
/home/ubuntu/lhy/run/Myproject/dataset/dataset3.gr
```

## 2.1 参数解释

dist-graph-convert: 数据集格式转换的可执行文件

edgeType: 指定边类型

bin2gr: 执行何种格式转化

tempDir: 临时目录, 中间需要放临时文件

numNodes: 顶点数

dataset3.bin: 转换前的数据集文件

dataset3.gr: 生成的数据集文件

## 3. 执行命令

```
GALOIS_DO_NOT_BIND_THREADS=1 mpirun -n=1
/home/ubuntu/1hy/run/build/lonestar/analytics/distributed/bfs/bfs-push-dist
/home/ubuntu/1hy/run/Myproject/dataset/dataset3.gr -t=1 -startNode=3 -
output=true -outputLocation /home/ubuntu/1hy/run/Myproject/output/
1>>/home/ubuntu/1hy/run/Myproject/output/bfs_push_log.txt
```

### 3.1 参数解释

GALOIS\_DO\_NOT\_BIND\_THREADS=1: 设置环境变量

mpirun -n=1: 使用mpi多进程, 在一个机器上使用的多进程数

bfs-push-dist: 执行命令的可执行文件

dataset3.gr: 图数据集文件

t=1: 在一台机器上使用的线程数, 不包括通信线程, 它被所有提供的分布式基准所使用。请注意, gpu只使用1个线程 (不包括通信线程)

startNode: 起始节点

output: 是否输出

outputLocation: 输出结果的文件地址

1>>: 将日志输出到文件