

# Check-in

Weixiong Zheng

August 8, 2018

## 1 Updates on previous goals

- Get the runtime error done so BART will be at where we restart
- See if there's anything I can help for Marissa

## 2 Progress up to now

### 2.1 Debugging progress and testing on Aug 01

From last meeting until last Friday, I was working on debugging on runtime error, which took much more time than what I expected. There are various types of things causing this. One of them is about templating. One of them, for instance, comes from using serial `Triangulation<dim>` for 1D and distributed one `parallel::distributed::Triangulation<dim>` for multi-D. This ends up making me separately template methods wherever triangulation shows up.

### 2.2 Debugging progress and testing on Aug 08

The main progress was after finishing the debugging, I've kept experimenting on MPI related data structure for gtest, which will extensively needed in later work. What we ended up getting is that a BART testing environment for easy use in MPI is created. Normally using GTest, we do

```
class TestClass : public ::testing::Test {...}
```

In situations with MPI using the newly created environment in BART, we do

```
class TestClassMPI : public btest::BARTParallelEnvironment {...}
```

The goodness is one does not need to provide a `void TearDown()` to finalize MPI in the testing class. The `btest::BARTParallelEnvironment` is implemented with MPI finalization functionality in its override of `::testing::Test::TearDown()`. Moreover, it provides a function to initialize MPI

```

class BARTParallelEnvironment : public ::testing::Test {
public:
    void MPIInit () {
        char** argv;
        int argc = 0;
        int err = MPI_Init(&argc, &argv);
        ASSERT_FALSE(err);
    }

    void TearDown(){
        int err = MPI_Finalize();
        ASSERT_FALSE(err);
    }
};

```

The resulting way of doing testing is super easy now with MPI. Beside that one does not need to provide TearDown, when overriding `::testing::Test::SetUp`, one only need to add one more line to initialize MPI compared to serial testing:

```

TestClassMPI::SetUp() {
    this->MPIInit();
    ... // whatever was planned to do
}

```

## 2.3 Marissa

### 2.3.1 Before Aug 01

Multiple iterations were done to elaborate the thesis including formulation modification, data display, result interpretation etc.

### 2.3.2 On Aug 08

The main work was to direct Marissa doing literature review and filling in proper references in the thesis.

## 2.4 What is planned next?

Talking to Josh today for the work separation and introduce the MPI gtest environment usage.

## 3 Goals/Things will be going on

- Discuss details of testing for all the class we have right now and separate the workload for us two. It's too much for me to do alone. Besides, it will be a good practice for Josh.

- Integrate the new material class with the restart
- Start the research purpose development.