

# Work Log for October 12th-30th

Logan Brown

October 30, 2014

## 1 Goals for the Week(s)

1. Sphere Packing code
2. What do we do for Dr. Gilchrist
3. PSUADE Module
4. Optimizer Module

## 2 Progress/Notes

### 2.1 Sphere Packing Code

The sphere packing module is complete. Here's a few notes

1. It uses a one dimensional array of length  $\text{population} * \text{parameters}$ .
2. For now, it works with a one-to-one model where each sphere pack algorithm writes to its own individual GA. This is largely because I can't find a succinct way for one GA to collect information from multiple Sphere Pack algorithms.
3. The GA needs to have Mutation added to it. The algorithm is just converging to distances around 1 or 2. It's still a convergence, which is cool.
- 4.

### 2.2 What can we do for Mike

### 2.3 PSUADE Module

#### 2.3.1 Picking up where we left off

Attempt 2: Go to `build/CMakeFiles/psuade-bin.dir/build.make` line 59, and add after `maing Psuade.cpp.o`,

```

/usr/bin/c++ $(CXX_DEFINES) $(CXX_FLAGS)
    -o /home/lbrown/iel-2.0/PSUADE/Src/Main/Psuade.o
    -c /home/lbrown/iel-2.0/PSUADE/Src/Main/Psuade.cpp
/usr/bin/c++ $(CXX_DEFINES) $(CXX_FLAGS)
    -o /home/lbrown/iel-2.0/PSUADE/Src/Main/IELPsuade.o
    -c /home/lbrown/iel-2.0/PSUADE/Src/Main/IELPsuade.cpp

```

The problem this generates is that cmake ignores the first set of commands, and instead uses the second set, so it never makes Psuade.cpp.o

Attempt 3: Change CMakeLists.txt

```

add_library (psuade-lib ${LIBRARY_TYPE} ${psuade_SRC} ${psuade_HDRS} ${PDF_SRC})
add_library (iel-lib ${LIBRARY_TYPE} "Src/Main/IELPsuade.cpp" ${psuade_SRC} ${psuade_HDRS})
add_executable (psuade-bin Src/Main/Psuade.cpp)

```

Run build as usual, then go to iel-2.0/PSUADE/build/CMakeFiles/iel-lib.dir/Src/Main to find the .o file. What to do with the .o? what's in the .o?

### 2.3.2 dummy hello function

A good idea to test for a successful build, putting a dummy DIEL hello function in the libPsuade.so

### 2.3.3 argc, argv

## 2.4 Optimizer Module

The optimizer functions are distinct .cpp files. It may be worthwhile to compile these as their own IEL library. The main problem is the command line arguments. They are pretty intricate.

## 3 Goals for next Week

1. Future Goal