Work Log for September

Logan Brown

September 25, 2014

1 Week of September 22nd-26th

1.1 Goals for the Week

1. Build a test module
2. Install PSUADE 1.7.2
3. ~~Look at readPsuadeIO function (replacing fscanf with tget?)~~
4. PSUADE Module Work
5. Test Ising model once Jason adds it to the DIEL
6. Look into Optimizer (Bobyqa?, can it optimize after running?)

1.2 Progress/Notes

1.2.1 Build a test module

Test module is build built, see star1:iel-2.0/LBrown/hello
I started with a basic hello world, and made changes.

1. Change main to the name of the module. In this case, the function was LBhello in LBhello.c
2. Change (int argc, char** argc) to (IEL_exec_info_t *exec_info)
3. Add the following includes:

```
#include <mpi.h>                                    /*these*/
#include "IEL.h"                                    /*are*/
#include "IEL_exec_info.h"              /*required to use the IEL functions*/
#include "tuple_comm.h"
#include "arrayList.h"
```

4. Compile as a library, including the IEL header files.

5. Driver Code. Writing a driver code is outside the purview of this weekly report.

6. Write a makefile. I'll include mine in the misc folder, but it will obviously need to be customized to other applications

I began attempting some basic Tuple Comms, but that fell flat.

1.2.2 Install PSUADE 1.7.2

Installed locally and on Star1. Running 'make test' in the build directory now passes all the tests, on both machines.

The install directions are in psuade-log/psuade.pdf

1.2.3 ~~Look at readPsuadeIO function (replacing fscanf with tget?)~~

1.2.4 PSUADE Module Work

Main/Psuade.cpp is a red herring. It only parses the command line arguments, which is something I'm not sure how we do, or if we even want to do it. If you call PSUADE with an input file, it runs "psuade->getInputFromFile", and then "psuade->run();"

Here are our main concerns, based on my last conversation with Kwai

1. Gutting main/run and replacing with a module

2. Passing command line argument (input file)

3. MPI_COMM_WORLD may be an obstacle, even in serial?

4. Shared Library Functions

5. Dakota?

```
int PsuadeBase::run() throw(Psuade_Stop_Exception)
{
   int   refineType, anaMethod, samMethod;
   char  *appName, inString[200];
   FILE  *fp;
   pData pAppFiles, pPtr;

   if (psuadeIO_ == NULL)
   {
      printOutTS(PL_ERROR, "PSUADE::run ERROR - no PsuadeData object.\n");
      exit(1);
   }

   psuadeIO_->getParameter("app_files", pAppFiles);
   appName = pAppFiles.strArray_[0];
```

```
   fp = fopen(appName, "r");
   if (fp != NULL)
   {
      fscanf(fp, "%10c", inString);
      if (!strncmp(inString, "PSUADE_IO",9)) useRSModel_ = 1;
      fclose(fp);
   }

   psuade_stop = 0;

   psuadeIO_->getParameter("method_refine_type", pPtr);
   refineType = pPtr.intData_;
   psuadeIO_->getParameter("ana_method", pPtr);
   anaMethod = pPtr.intData_;

   if (refineType == 1 && anaMethod == PSUADE_ANA_ONESIGMA)
        runAdaptiveErrBased0();
   else if (anaMethod == PSUADE_ANA_ARSM)
        runAdaptiveNN();
   else if (anaMethod == PSUADE_ANA_ARSMMB ||
            anaMethod == PSUADE_ANA_ARSMMBBS)
   {
      psuadeIO_->getParameter("method_sampling", pPtr);
      samMethod = pPtr.intData_;
      if (samMethod == PSUADE_SAMP_METIS)
           runAdaptiveErrBased1();
      else runAdaptiveErrBasedG();
   }
   else if (anaMethod == PSUADE_ANA_REL)
        runAdaptivePRA();
   else if (anaMethod == PSUADE_ANA_AOPT)
        runAdaptiveOpt();
   else if (anaMethod == PSUADE_ANA_GLSA)
        runAdaptiveGradBased();
   else runUniform();

   if (psuadeIO_ != NULL) psuadeIO_->processOutputData();
   return 0;
}
```

1.2.5 Test Ising model once Jason adds it to the DIEL

1.2.6 Look into Optimizer (Bobyqa?, can it optimize after running?)

1.3 Goals for next Week

1. PSUADE 1.7.2 on Darter
2. PSUADE Module that runs as usual
3. Figure out Command Line arguments for DIEL
4. Find out what's going on with DAKOTA calling PSUADE