ParadisEO – PEO : Lesson 5

Note : All the components are not presented in this lesson (binary, topology, asynchronous or synchronous ...). To know the completeness of components refer to API documentation of ParadisEO – EO and ParadisEO – PEO.

Problem

In the lesson 5 you can execute an algorithm with an hybridization.

The problem is the traveling salesman problem (TSP) .

example.cpp :

```
// Libraries of problem
#include "param.h"
#include "route_init.h"
#include "route_eval.h"
#include "order_xover.h"
#include "edge_xover.h"
#include "partial_mapped_xover.h"
#include "city_swap.h"
#include "part_route_eval.h"
#include "merge_route_eval.h"
#include "two_opt_init.h"
#include "two_opt_next.h"
#include "two_opt_incr_eval.h"

#include <peo>
#define POP_SIZE 10
#define NUM_GEN 10
#define CROSS_RATE 1.0
#define MUT_RATE 0.01
int main (int __argc, char * * __argv)
{
// Example of a problem (TSP) with an hybridization
  peo :: init (__argc, __argv);
  loadParameters (__argc, __argv); // Processing some parameters relative to the
                                   // tackled problem (TSP)
  RouteInit route_init; // Its builds random routes
  RouteEval full_eval; // Full route evaluator
  OrderXover order_cross; // Recombination
  PartialMappedXover pm_cross;
  EdgeXover edge_cross;
  CitySwap city_swap_mut;  // Mutation
// Local Search
  TwoOptInit pmx_two_opt_init;
  TwoOptNext pmx_two_opt_next;
  TwoOptIncrEval pmx_two_opt_incr_eval;
  moBestImprSelect <TwoOpt> pmx_two_opt_move_select;
```

```
  moHC <TwoOpt> hc (pmx_two_opt_init, pmx_two_opt_next, pmx_two_opt_incr_eval,
pmx_two_opt_move_select, full_eval);
// The EA
  eoPop <Route> ox_pop (POP_SIZE, route_init);  // Population
  eoGenContinue <Route> ox_cont (NUM_GEN); // A fixed number of iterations
  eoCheckPoint <Route> ox_checkpoint (ox_cont); // Checkpoint
  peoSeqPopEval <Route> ox_pop_eval (full_eval);
  eoStochTournamentSelect <Route> ox_select_one;
  eoSelectNumber <Route> ox_select (ox_select_one, POP_SIZE);
  eoSGATransform <Route> ox_transform (order_cross, CROSS_RATE, city_swap_mut,
MUT_RATE);
  peoSeqTransform <Route> ox_para_transform (ox_transform);
  eoEPReplacement <Route> ox_replace (2);
  peoEA <Route> ox_ea (ox_checkpoint, ox_pop_eval, ox_select, ox_para_transform,
ox_replace);
  ox_ea (ox_pop);   // Application to the given population
  peo :: run ();
  peo :: finalize (); // Termination
  if(getNodeRank()==1)
      std :: cout << "\nResult : "<<ox_pop[ 0 ].fitness();
  hc( ox_pop[ 0 ] );
  if(getNodeRank()==1)
      std :: cout << "\n\nAfter an hybridization : " << ox_pop[ 0 ].fitness();

}
```

Launching the program

Your file should be called example.cpp - please make sure you do not rename the file (we will be using a pre-built makefile, thus you are required not to change the file names). Please make sure you are in the paradiseo-peo/tutorial/build/Lesson5 directory - you should open a console and you should change your current directory to the one of Lesson5.

Compilation :

- make
- make install

Execution (ie Technical Introduction):

mpiexec -n 4 ./example @param