Fixing the port-error issue in implementing Island Model in ECJ

Abdul Wazed Chowdhury, Graduate Student, Computer Science, Brock University

February 20, 2014

ECJ-version used: 20

In the island model of parallelization, the computational workload is distributed into multiple GP processes, running on separate or the same physical machine. Each of the processes (also called islands) evolves its own population and occasionally exchanges few fit-individuals with the other islands immediately before breeding or accepts new individuals sent by other islands immediately after breeding.

The problem:

ECJ has a great support for island model GP. In ECJ, you can configure a bunch of island-model parameters very easily and efficiently, examples include topology of the island-model, number of islands, server as a separate process, making an island the server, exchange rate between islands individually, exchange interval, synchrony, mailbox capacity, immigrant selection criteria, client-server communication, so on and so forth.

But unfortunately all of us were facing a strange problem at the beginning of our implementation of island model in ECJ. The evolutions of the islands were being stopped after certain runs, only 1 run in my case. It was varying from machine to machine, however, one of us was able to have 12 successful runs in a row, although he did set parameters for 20 runs. Here is a snapshot of the error that appeared at the beginning of the second run (Job#0 is the first run) in my machine; however, I was using one of the islands as the server then.

```
.....
Job: 1
Setting up
Processing GP Types
Processing GP Node Constraints
Processing GP Function Sets
Processing GP Tree Constraints
Initializing Generation 0
The communication will be asynchronous.
Server Launched.
Connecting to Server 192.168.1.111, port 8991
FATAL ERROR:
Error creating a socket on port 8991
Java Result: 1
BUILD SUCCESSFUL (total time: 2 minutes 13 seconds)
```

Bearing in mind that there may be other solutions to this very same problem, I would like to appreciate if anybody finds a bug in this particular solution and provide a corresponding fix. For this solution, I will be using a separate ec.exchange.IslandExchange process as the server.

Server part:

After a through debugging, it was found that the server process was exiting immediately after each run and as a result the clients were unable to reconnect to it at the beginning of the next run(s). I have tried different port numbers, but nothing good happened. I figured it out by placing the block of code responsible for starting the server into a loop where the server started 20 times (equal to the number of runs), each time using the next port number. Also, the clients were provided the initial server-port number and were configured to connect to the server in the consecutive port numbers in the following runs. To keep track of this, two class variables were declared inside IslandExchange.java.

```
    static boolean initializeServerPort=true;
    public static int serverPortNew;
```

Here is the while loop that starts the server at the beginning of each run, with a new port of course. As you will find, this loop is to be placed inside the main() of IslandExchange.java.

```
int count=1;
while(count<=numberOfJobs){
    ie.setup(myEvolutionState,myBase);
    ie.fireUpServer(myEvolutionState,myBase);
    ie.serverThread.join();

    output.flush();
    System.err.flush();
    System.out.flush();
    serverPortNew++; // initialize the serverPortNew before entering the loop
    count++;
}
```

Here, the ie.setup() method is responsible for things like reading serverPort from server.params, detecting current IP address etc.
ie.fireUpServer() creates an instance of the IslandExchangeServer which is actually a thread that implements the runnable interface. Inside the setupServerFromDatabase() method of this class we will add the lines to initialize serverPortNew variable and to prohibit further initialization, we will also set initializeServerPort to false.

```
serverPort = state.parameters.getInt( p, null, 1 );
if(IslandExchange.initializeServerPort==true){
    IslandExchange.initializeServerPort=false;
    IslandExchange.serverPortNew=serverPort;
}
```

Inside the run method of IslandExchangeServer, the server thread is actually created against a port and we will use our serverPortNew while creating the socket.

```
serverPort=IslandExchange.serverPortNew;
try
    {
    serverSocket = new ServerSocket(serverPort,numIslands);
    }
catch ( IOException e )
    {
```

```
    state.output.fatal( "Error creating a socket on port " + serverPort );
    }
```

To avoid further trouble, the original instance/class variables (serverPort, for example) were left untouched.

Client/Island part:

As mentioned before, the islands were informed the initial server-port number through the parameter file and at the beginning of subsequent runs each island was configured to look for the server in the consecutive port numbers. For example, if the server starts with the port number 8265, at the beginning of the second run, it will switch to 8266, in the third run, 8267, so on and so forth. Now, for an island to do that, a similar technique was applied. A static variable was declared which after the initialization with first server-port number, was just incremented accordingly. As the describe() method is called at the end of each run, it will be a good idea to increment the server-port counter for islands at the bottom of the describe method. However, the same effect may be achieved in a number of other ways.

Server output:

Island Exchange Server
Used in ECJ by Sean Luke
IP ADDRESS: 192.168.1.107
The communication will be synchronous.
Island Kutubdia logged in
0: Island Kutubdia has address 192.168.1.107 : 9002
Island Maheshkhali logged in
1: Island Maheshkhali has address 192.168.1.107 : 9004
Island Sandwip logged in
2: Island Sandwip has address 192.168.1.107 : 9006
Island Hatiya logged in
3: Island Hatiya has address 192.168.1.107 : 9008
Island Kutubdia should connect to island Maheshkhali at 192.168.1.107 : 9004
Island Kutubdia should connect to island Sandwip at 192.168.1.107 : 9006
.....
Server: Island Kutubdia dropped connection
Server: Island Sandwip dropped connection
Server: Island Hatiya dropped connection
Server: Island Maheshkhali dropped connection
Server Exiting
IP ADDRESS: 192.168.1.107
The communication will be synchronous.
Island Maheshkhali logged in
0: Island Maheshkhali has address 192.168.1.107 : 9004
Island Hatiya logged in
1: Island Hatiya has address 192.168.1.107 : 9008

Island Sandwip logged in
2: Island Sandwip has address 192.168.1.107 : 9006
Island Kutubdia logged in
3: Island Kutubdia has address 192.168.1.107 : 9002
Island Maheshkhali should connect to island Kutubdia at 192.168.1.107 : 9002
Island Maheshkhali should connect to island Sandwip at 192.168.1.107 : 9006

Client output:
Job: 0
Setting up
Processing GP Types
Processing GP Node Constraints
Processing GP Function Sets
Processing GP Tree Constraints
Initializing Generation 0
I'm just a client.
Connecting to Server 127.0.0.1, port 8265
Connected to Server after 0 ms
IslandExchangeMailbox created.
My address is: 192.168.1.107
The communication will be synchronous.
Trying to connect to 192.168.1.107 : 9004
Trying to connect to 192.168.1.107 : 9006
Island Maheshkhali connected to my mailbox
Trying to connect to 192.168.1.107 : 9008
Island Hatiya connected to my mailbox
Island Sandwip connected to my mailbox
All islands have connected to my client.
Subpop 0 best fitness of generation: Fitness: Standardized=105032.0 Adjusted=9.520817E-6 Hits=1368
Waiting for synchronization....
Generation 1
Subpop 0 best fitness of generation: Fitness: Standardized=105032.0 Adjusted=9.520817E-6 Hits=1368
.....
Job: 1
Setting up
Processing GP Types
Processing GP Node Constraints
Processing GP Function Sets
Processing GP Tree Constraints
Initializing Generation 0
I'm just a client.
Connecting to Server 127.0.0.1, port 8266
Retrying
Connected to Server after 100 ms

IslandExchangeMailbox created.
My address is: 192.168.1.107
Island Maheshkhali connected to my mailbox
The communication will be synchronous.
Trying to connect to 192.168.1.107 : 9004
Island Hatiya connected to my mailbox
Island Sandwip connected to my mailbox
All islands have connected to my client.
Trying to connect to 192.168.1.107 : 9006
Trying to connect to 192.168.1.107 : 9008
Subpop 0 best fitness of generation: Fitness: Standardized=105030.0 Adjusted=9.520999E-6 Hits=1370
Waiting for synchronization....
Generation 1
Subpop 0 best fitness of generation: Fitness: Standardized=105023.0 Adjusted=9.521633E-6 Hits=1377
Waiting for synchronization....

Island as the server:

If you plan to use one of your islands as the server, here is a fix of the same problem provided by Ashkan Entezari, another graduate student of Computer Science.

Hi,
What I did was trying to change the server port, here is what I exactly did:

in *ec > exchanger.java* define this static variable like this:

```
public static int cnt = 0;
```

in *ec.exchange > IslandExchanger.java* go to setup() function and there should be the following line (I think it's line 490):

```
serverPort = state.parameters.getInt( p, null, 1 );
```

now put the following code beneath this line:

```
cnt++;
serverPort+=cnt;
```

then in the same file find the following code (it should be the line 1708):

```
serverPort = state.parameters.getInt( p, null, 1 );
```

put the following code after this line:

```
serverPort+=IslandExchange.cnt;
```

Hope that helped.