Project Arimaa – CR9 : Wednesday, November, 12th

Presents : Everyone, with Nikos Parlavantzas and Christian Raymond

Order of business :

1. MinMax controversy (solved)
2. CAF over MPI
3. Comments on plan of the second report
4. I/O software (Dan)
5. Questions about report

Information :

- There was a controversy about the composition of the algorithm of MCTS. It was about the treatment of the node, how to go up in the trees, updating statistics. It was either the Minmax version (take the best pourcentage, and substract it to 100 to get the previous node) and the MCTS version (add wins and views). We used Wikipedia, and the teachers to decide which one was MCTS and Minmax. A good way to understand is the picture explaining the calculus on Wikipedia MCTS.
- MPI is not really optimized to do it, the code is more complicated to do asyncronous communications.
  CAF is better on asyncronous, the question was "do we use a computer as a thread, or a cluster", which is more high level in MPI. The answer of the teacher was it was possible to do it.
  There is less documentation in CAF but better optimized. It works with IP and computers, we should have no problem, only some difficulties. We just need to test it and do a prototype.
  MPI is a big community, but CAF have a more high level. Another problem is we can encounter an error not present in the documentation, which will be terrible to correct (send a message to the community seems to be the best to do) CAF will replace OpenMP and MPI, and supports GPU. All people seems to give their approval to CAF but we'll make our choice later, maybe just a comparision between the two.
- Input/Output software will be changed. Nikos wanted to make communicate two computers with each other, which is out of our initial topic. We only need to do it on a single computer, but make two algorithms to compete with each others. They will work one by one, using all resources of the computer.
- This part will only be an interface and link towards with a game client, handling bots maybe. But this will be done only if we have time. For now, it will be an interface implemented with MCTS methods/Normal player method. If we have time, we'll implement it with bot methods.

Planification

Task	Responsible	Deadline
OpenMP report part	Benoit and Baptiste	11/19
MCTS report part	Benoit	11/19
MPI report part	Mikail	11/19
Parallelization method report part	Mikail	11/19
OpenAcc report part	Baptiste	11/19
Behaviour of the game report part	Gabriel	11/19
General Architecture review	Gabriel	11/19
Conclusion report part	Gabriel	11/19
Input/Output report part	Dan	11/19
Introduction report part	Dan	11/19

Abstract report part	Prateek	11/19
API report part	Prateek	11/19
First draft of the report	Everyone	11/19
Last draft of the report	Everyone	11/26
Due date for the report	Dan	11/27
OpenMP implementation	Not decided	Begin after 11/19
Finish the game application	Gabriel	12/18

Next meeting : 11/19