1. Projet et objectif
2. Performances de départ
3. Modifications apportées
4. Problèmes rencontrés
5. Performances finales
6. **Projet et objectif**

-Tom Kerrigan's Simple Chess Program (TSCP) 1997 C++

-Optimisation du code et donc amélioration des performances

1. **Performances de départ**

Node :

Temps d’exécution :

Node/s :

1. **Modifications apportées**

Optimisation 2 en suivant guideline :

-Ajout can\_attack[6][64][64]

-Pas de meilleure optimisation par l’utilisation de bitboards et de masques (après si on finalise déjà l’optimisation 2)

-Modification de attack()

-Ajout initAttackTables()

-Vérification par memset ??? Même nombre de nodes => pb pas assez de tests intermédiaires pour le vérifier

-Ecriture initAttackTables : remplis can\_attack de 0,

|  |
| --- |
| //memset( canAttack,1,sizeof(canAttack) ); // Pour test  for(int p =0;p<6;p++){  for (int i=0;i<64;i++){  for (int j=0; j<64;j++ ){  can\_attack[p][i][j]=0;  }  }  }  for(int p =0;p<6;p++){  for (int i=0;i<64;i++){  for (int j = 0; j < offsets[p];j++){  for (int n = i;;) {  n = mailbox[mailbox64[n] + offset[p][j]];  if (n == -1){  break;  }  can\_attack[p][i][n]=1;  if (!slide[p]){  break;  }  }  }  }  } |

1. **Problèmes rencontrés**

Temps de compréhension conséquent du programme TSCP complexe.

Abandon optimisation 1, confusion + trop de code pré-existant à changer.

Optimisation OK mais diminution nombre de node => retour sur projet initial

Non utilisation des tests asserts

1. **Performances finales**

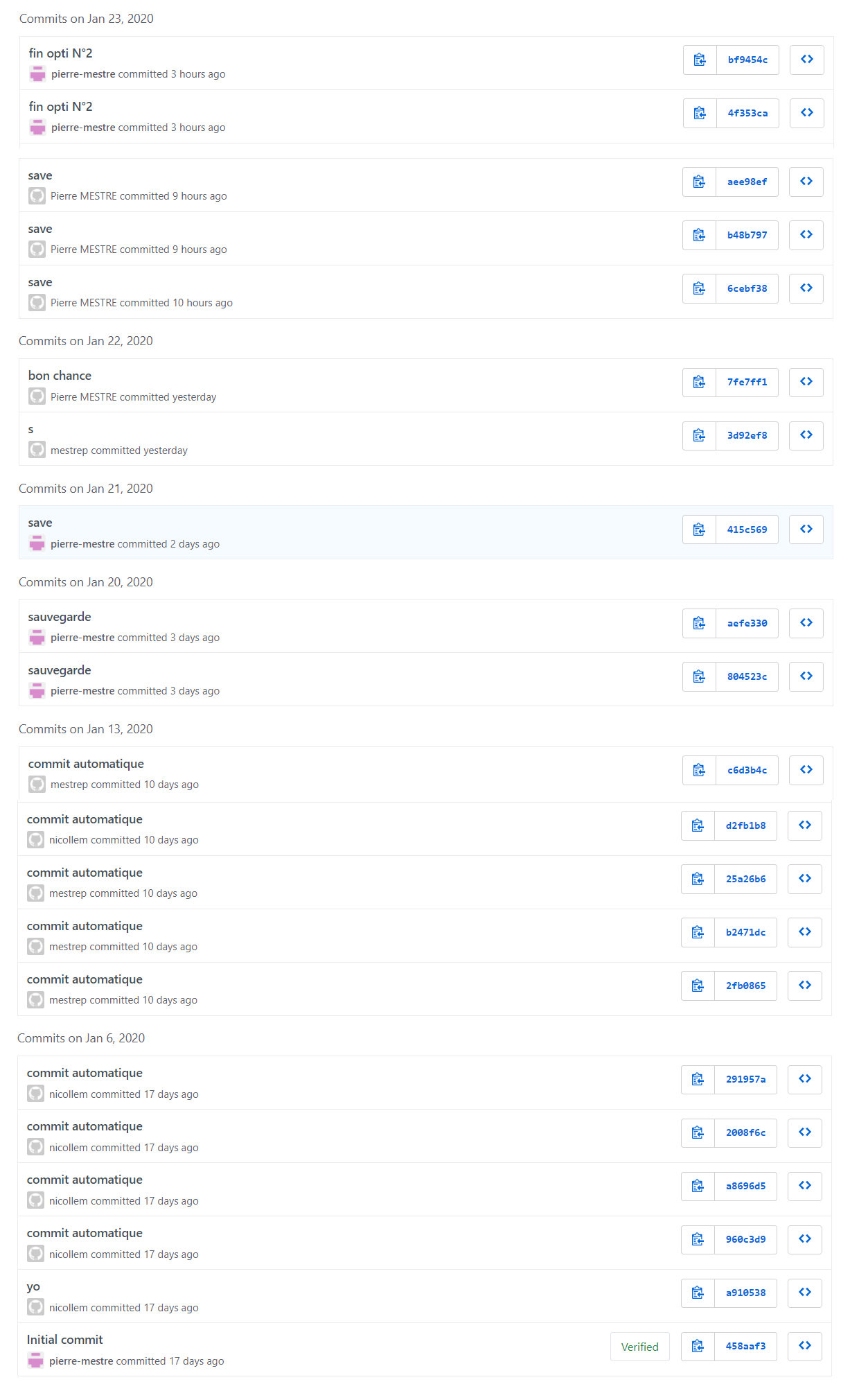
Node :

Temps d’exécution :

Node/s :

Le nombre de node analysés au final est le même qu’initialement, donc il n’y a pas eu de perte à ce niveau là, ni de bugs. Le temps d’exécution à été diminué puisque le programme analyse maintenant plus de node par seconde. Le programme a donc bien été optimisé.

Comparaison avec perf ini ? Pourcentage d’amélioration ?



Check si les commits se font bien

Récupération du programme TSCP original

Création du fichier .txt ID et MDP

Création du script de mise à jour

Renomme le fichier .txt ID et MDP

Ajout commentaires + fichiers Winboard + premiers fichiers de log de Debug

Ajout Sublime Text

Màj script git + configuration Sublime Text

Màj script git

Configuration Winboard + sauvegarde «Ordi VS Nous»

Màj Sublime Text + **(1) Ajout tab pospieceW, pospieceB, piece\_dead**

1. **Ajout de des «useful squares» (defs.h)**
2. **Correction des «useful squares»**

**(4) Modif tab pospieceW, pospieceB + ajout posIniPieceW et posIniPieceB**

**(5) Modif PIECE\_DEAD + parcours des tab pospieceW et pospieceB dans init\_board()**

**(6) Modif init\_board() + modif attack() (copie ORIGINAL + changement for et PAWN pour 3 <= i <= 7)**

Configuration Sublime Text **+ (7) Ajout initAttackTables() pour test + ajout canAttack[6][64][64] data.h + Modif des des «useful squares»**

**(8) Modif initAttackTables() + petites modif tab can\_attack**

**(9) Modif des «useful squares» + ajout tests assert**

**(10) Retire commit (9)**

**(11) Modif projet ???**